Data mining techniques for complex application domains

POLITECNICO DI TORINO

PhD in Information and System Engineering
XXV cycle

III Facoltà di Ingegneria
Settore scientifico ING-INF/05

PhD Thesis

Data mining techniques for complex application domains

Author:
Naeem Ahmed Mahoto
Matr. 169222

Supervisor:
Prof. Elena Baralis
Co-supervisor:
Prof. Silvia Chiusano

March 2013
A.A. 2012/2013

Dedicated to my family

Acknowledgments

All praise be to Allah, Lord of the worlds, the Beneficent, the Merciful, Who blesses us with His kindness, mercy and help that made me capable to accomplish my studies.

I feel great privilege to express my deep gratitude to my supervisor Professor Elena Baralis for providing me with the opportunity to work in the research in data mining, for her kind interest, valuable suggestions, encouragement and guidance that leaded me at all levels.

I would like to give explicit credit to Professor Silvia Chiusano for her valuable tips and suggestions that made me able to focus on right things for the research. In addition, I am also grateful to all my research group colleagues Alessandro Fiori, Tania Cerquitelli for their suggestions at times, and specially Luca Cagliero for his help, and kind suggestions.

A very special thank is due from me to Giulia Bruno who guided, helped, and supported me at every step. It has been a pleasure working with these nice and generous people.

Contents

Chapter 1

Introduction

The emergence of advanced communication techniques has increased availability of large collection of data in electronic form in a number of application domains including healthcare, e-business, and e-learning. Everyday a large amount of records are stored electronically. However, finding useful information from such a large data collection is a challenging issue. The data mining techniques are greatly adopted to retrieve valuable and interesting knowledge from such a huge amount of data. Data mining technology aims automatically extracting hidden knowledge (i.e., valuable patterns) from large data repositories exploiting sophisticated algorithms, just like a miner uses various tactics depending on the earth surface and outer environment to mine gold, iron, and other valuable metals from earth. The extracted information is significantly valuable in many application domains including business, education, science, and healthcare systems. Knowledge discovery in databases (KDD), comprising of a series of steps, converts raw data into meaningful information [153]. Although several definitions are given for data mining and KDD, some as examples are described in the following:

- *"Knowledge discovery in databases is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data."*[53]
- *"Data mining is finding hidden information in a database."*[45]
- *"Data mining is the process of extracting valid, previously unknown, comprehensible, and actionable information from large databases and using it to make crucial business decisions."*[139]

- *"Data Mining is the process of automatically discovering useful information in large data repositories."*[153]

The KDD process consists of the following main phases as illustrated in Fig.1.1: data collection, data selection, data preprocessing, data transformation, data mining, and knowledge interpretation.

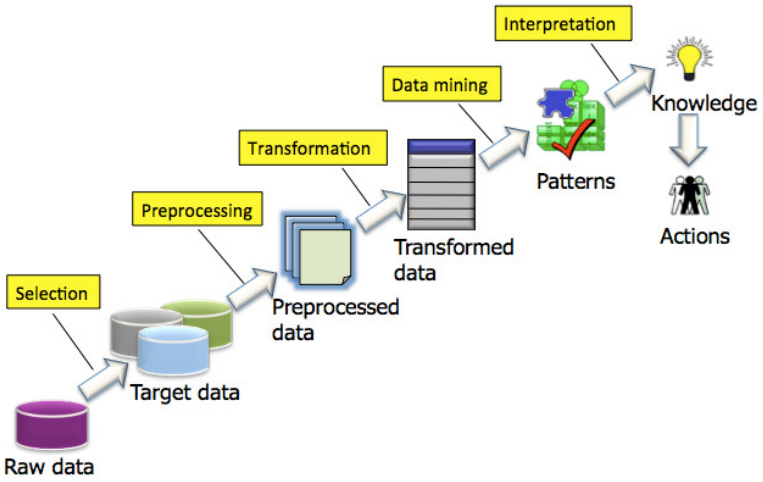


Figure 1.1: The KDD process [54]

Data collection. Data is collected from different units into a warehouse for the data mining process.

Data selection. Domain experts are involved at this phase to select the fields/attributes that are related to the problem.

Data preprocessing. The preprocessing phase cleans the dataset by removing erroneous and noisy data and providing missing values. This phase is also very much necessary for the final results because incorrect and erroneous data may lead towards incorrect information.

Data transformation. At this stage, the preprocessed data is transformed into suitable format so that mining algorithms can be directly applied.

Data mining process. This phase normally uses different algorithms and techniques to get useful knowledge. However, certain algorithms can be used for some specific problem domains.

Information evaluation/interpretation. Domain experts interpret the mined patterns. The extracted information are then applied in the real world applications based on its correctness, completeness, and novelty.

Generally, data mining tasks are classified into two categories: descriptive and predictive. Descriptive mining emphasizes on general properties of the data in database. Predictive mining infers some predictive clues based on the current data in database [66]. Different data mining techniques are applied to get targeted information.

The hidden knowledge in the electronic data may be potentially utilized to facilitate the procedures, productivity, and reliability of several application domains. For example, in the healthcare context, the practitioners, pharmaceutical personnel and medical staff can be provided guidelines by exploiting data mining techniques to enhance medical treatment procedures, optimal use of resources, and significantly lower time and cost. The online textual data may be found in either plain textual format and blogs or user-generated content (i.e., tweets or user comments on social networking). Whilst, the hidden knowledge inside textual data, may help to provide an insight information about user behaviours, web services to support management and decision making strategies.

Data mining relies on databases containing raw data, thus it faces several difficulties. The major issues, which may be contained in database, can be incomplete data (missing values) and incorrect data (noisy data). Apart from these issues, the database may be limited to some specific application domain at limited scope, which in return cannot express the generalized outcomes for the domain. The problems and pitfalls are not only concerned with outcomes of data mining techniques but also fall into inadequate technological resources. For example, an application that results quick and correct outcomes based on small training datasets, may behave partially or completely different while dealing with large amount of database. Similarly, small dataset may fit into memory space where as larger database may cause insufficient memory errors [33].

General purpose data mining approaches are often unsuitable for addressing advanced analysis on complex domains. Data mining solution must be tailored to the problem under analysis to support domain experts in discovering fruitful knowledge.

The PhD activity has been focused on novel and effective data mining approaches to tackle the complex data coming from two main application domains: *Healthcare data analysis* and *Textual data analysis*.

Healthcare data analysis. Since Healthcare problems are evolutionary, complicated, and diversified, there is the need of automatic data discov-

ery to enhance medical. For example, the analysis of correlations between various symptoms for a given disease and data regarding resource utilization can provide information to enhance the healthcare services. The extracted knowledge leads healthcare systems to achieve profitable, effective, significantly accurate treatment procedures, and may possibly reduce costs. Moreover, healthcare data is diversified due to the high dimensionality of the data comprising of medical history records of patients having different symptoms. Besides, being health an essential aspect of life, efficient, sophisticated, and correct techniques and procedures are highly required to address healthcare data problems. However, since the manual analysis of vast amount of patients' records is complex task, thus, novel data mining techniques are needed to utilize resources effectively, reduce cost, save time, and identify non-compliant processes timely.

The PhD research activity, in the context of healthcare data, addressed the application of different data mining techniques to discover valuable knowledge from real exam-log data of patients. In particular, efforts have been devoted to the extraction of medical pathways, which can be exploited to analyze the actual treatments followed by patients. The derived knowledge not only provides useful information to deal with the treatment procedures but may also play an important role in future predictions of potential patient risks associated with medical treatments.

Textual data analysis. Textual data is commonly available in electronic documents or on social networks. Its peculiar features prevent the application of traditional data mining techniques. Hence, proposing novel data mining approaches to discovery of valuable knowledge from such kind of data is a challenging task.

The research effort in textual data analysis is twofold. On the one hand, a novel approach to discovery of succinct summaries of large document collections has been proposed. On the other hand, the suitability of an established descriptive data mining technique (i.e., generalized association rule mining [143]) to support domain experts in making decisions has been investigated. Both research activities are focused on adopting widely exploratory data mining techniques to textual data analysis, which require to overcome intrinsic limitations for traditional algorithms for handling textual documents efficiently and effectively.

This thesis is organized as follows. Chapter 2 describes the data mining techniques adopted in healthcare domain. This chapter presents the approaches proposed to extract medical pathways from real healthcare data

of patients in three different pathologies considered as case studies. Chapter 3 introduces the text mining and presents the adopted approaches in multi-document summarization and investigation of user-generated content. Furthermore, the previous works on text summarization as well as on the analysis of social network data; the experimental results from real-world data and performance of the proposed approaches are also reported. Finally, Chapter 4 draws conclusions and discusses the future developments for the proposed approaches.

Chapter 2

Healthcare Data Mining

This chapter describes data mining techniques for healthcare domain. The knowledge discovery from healthcare data is presented in Section 2.1 and the related research work in Section 2.2. The proposed approaches for the extraction of medical pathways are reported in Section 2.3, while for exam correlations in Section 2.4 and Patient clustering in Section 2.5.

2.1 Knowledge discovery from healthcare data

The introduction of electronic medical records has made available a large amount of medical data, storing the medical history of patients. This large data collection can be profitably analyzed by using data mining techniques to extract a variety of information, for example the relationships between medical treatments and final patient conditions, or the medical protocols usually adopted for patients with a given disease [26]. Data mining focuses on studying effective and efficient algorithms to transform large amounts of data into useful knowledge [153].

An actual problem in this domain is to perform reverse engineering of the medical treatment process to highlight medical pathways typically adopted for the specific health conditions, as well as discovering deviations with respect to predefined care guidelines. The medical pathway is a course of treatment performed by a certain patient to cure a certain disease, and is prescribed by the medical experts. The standard medical pathways define care guidelines for a number of chronic clinical conditions, specifying the course of treatment (i.e., sequences or patterns) and timing of actions necessary to perform for a given disease in an effective and efficient way. Thus,

the discovered knowledge can support healthcare organizations in improving the current treatment processes or assessing new guidelines.

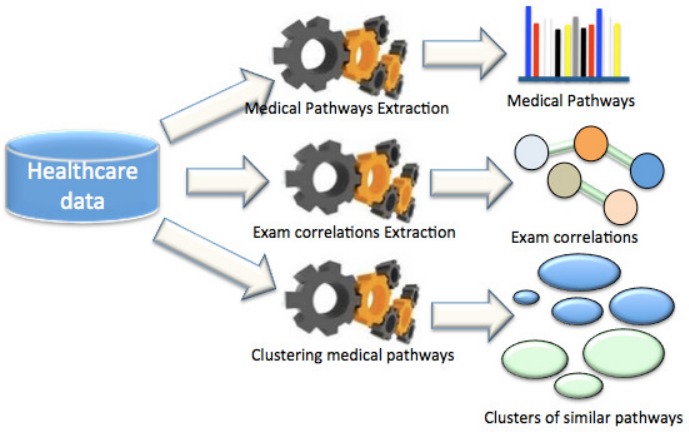


Figure 2.1: Knowledge extraction process

The PhD research activity aims at extracting useful knowledge from real healthcare data. The knowledge extraction process is carried out by means of well established data mining techniques such as sequential pattern mining [4], association rule mining [2] and clustering techniques [153] as presented in Fig. 2.1. The proposed approaches are described in details later in this chapter.

2.2 Related works

Many healthcare applications are developed for enhancement of the healthcare management as well as patients' care [1]. Everyday, a large collection of medical data are stored in electronic format in healthcare organizations. The analysis of such an immense amount of data is a complex and very difficult task, since each patient has his/her medical history and disease complexities and conditions. Therefore, the investigation of medical data is an open and challenging issue. To cope with this problem, data mining techniques have taken great attention to investigate large collection of medical data [67]. For example, [26] demonstrated medical care enhancement, reduction in time and cost from large medical data records. The decision support tools in healthcare exploiting data mining techniques are reported in [84, 116, 138], however, none of them worked on real dataset for the experiments.

Data mining studies are recommended to carry on in medical field by [15]. Furthermore, well established sequential pattern mining [4] technique has been successfully exploited for analyzing electronic medical data. [147] reported data warehousing and data mining techniques essential for the provision of evidence-based guidelines for clinicians. The work in [160] promoted data warehousing and data mining activities for clinical and administrative data in healthcare management. [148] presented prediction model to predict pressure ulcer development. The model exploits data mining techniques namely Mahalanobis Taguchi System (MTS) [150], Support Vector Machines (SVMs) [158], Decision Tree [120] and Logistic Regression (LR) [72] to predict incidence of pressure ulcers, which helps to take care of surgical patients and reduce occurrences of pressure ulcers; thus ensuring good health. Moreover, MTS performs better in comparison with other techniques applied in the research [147].

Data mining techniques such as association rules [2], clustering [153], sequential pattern mining [4] and classification [153] are essential in building the new healthcare applications. These sophisticated algorithms are applied to obtain relevant information from medical data. The extracted information provides a number of benefits such as helping physicians, deeply investigation of treatment processes and helping in building of standard care guidelines [6]. The literature about exploitation of different data mining techniques in healthcare data is reported in the following.

2.2.1 Medical pathways mining

A number of research is carried out for enriching and refining clinical guidelines to support medical pathways (also clinical pathways) and practices, disease management and resource utilization [30, 73, 76]. Clinical pathway mining is an essential aspect aiming at discovery of medical patterns [73]. Standard medical pathways, also termed as clinical pathways, are defined as care guideline for a variety of prolonged clinical conditions. These pathways postulate the necessary sequence and timing of actions to offer treatments to patients. The adoption of such guidelines help healthcare management to control both their treatment processes as well as costs [23, 117]. The clinical pathways and clinical guidelines may differ from each other due to administrative reasons; clinical guidelines require consensus between medical experts, however, clinical pathways imitate a-priori treatment of patients [5, 73, 160].

The discovery of time dependency patterns of clinical pathways to manage

brain strokes are reported in [92]. The obtained patterns help in prediction of patients´admission in hospitals. A data mining framework for finding alternative practices and evaluation of liver disease is reported in [136]. The prototype was demonstrated in Taiwan and Mangolia. However, using cloud computing [29] and Service Oriented Architecture (SOA) [49], the framework could be shared in other countries. A role of predictive data mining in clinical medicines is discussed in [15]. The use of semantic web technologies, precisely ontologies and semantic rules for integrating clinical pathways in clinical decision support system is reported in [171].

A variety of techniques for clinical pathways analysis is proposed. These techniques help to either redesign or optimize the existing clinical pathways [132]. Among such techniques clinical statistical pathway analysis can also be exploited for investigation of medical pathways, for instance, statistics of pathway abort rate and pathway coincidence rate [46, 166]. The medical experts interpret the set of patterns of several patients to deeply analyze treatment procedures [73]. The mining of closed clinical pathways from clinical workflow logs, which record medical behaviours as events in patients' course of treatment is described in [73]. The work in [73] detects most of the regular frequent clinical pathways given a clinical workflow log. However, the infrequent pathways, which are variants and/or missing in extracted pathways need to be discovered and properly analyzed.

The frequent sequence extraction from exam log data for analysis of medical pathways of diabetic patients is proposed in [10]. A self-learning expert system for identifying symptoms in Traditional Chinese Medicine by means of data mining techniques, particularly, using improved hybrid bayesian network learning algorithm is discussed in [164]. The proposed system in [164] is data driven in nature and constructs knowledge base by learning automatically from clinical data and the knowledge of experts. It differs from rule-based systems that inherent knowledge through interactions between experts and knowledge engineers.

2.2.2 Association rule mining in healthcare data

The association rule mining is greatly exploited in medical domain for analyzing the relationships between various symptoms for a given disease. The possible side effects of using multiple drugs during pregnancy period are presented in [31] using association rule mining [2] approach. [31] used SmartRule technique (spreadsheet software) to mine association rules from a stored tabular pregnancy data and achieved Maximum Frequent Itemsets (MFI) for a

user given minimum support threshold. The association rules are obtained with targeted attributes from a selected subset of MFIs. Further, the resultant rules are arranged in hierarchical tree structure. The work in [31] highlighted and warned about the drugs that may cause harm to unborn babies. The impacts in management strategies against human immune-deficiency virus (HIV) are addressed in [62] using decision tree and association rule mining techniques. Several data mining techniques are addressed including diagnosis of heart disease [137] and prediction of heart attacks in [1, 144].

A bridging rule concept is proposed in [173], which is a kind of outlier and a new type of patterns. Further, [173] investigated associations between two or more conceptual clusters. The antecedent and consequence of bridging rule contained in different clusters. For example, consider two clusters C_a and C_b, each of the cluster comprises of some data objects, e.g., $C_a = \{c_{a1}, c_{a2}, c_{a3}, ..., c_{an}\}$ and $C_b = \{c_{b1}, c_{b2}, c_{b3}, ..., c_{bn}\}$. The data objects of each cluster may be having relationships among themselves. A rule $c_{a1} \Rightarrow c_{b1}$ is a bridging rule, since its antecedent and consequence belong to different conceptual clusters. A bridging rule is similar to association rule but it differs in two reasons:
i. *It can also be extracted from infrequent patterns*
ii. *It is measured by its importance*
The bridging rule helps to discover information that might have been missed during clustering and classification of the concepts [173].

2.2.3 Clustering techniques in healthcare data

Clustering algorithms are widely adopted in several medical applications for different viewpoints. An approach for discovery and integration of frequent sets of features from distributed databases is presented in [43] by means of unsupervised learning (i.e., hierarchical clustering [81]). Precisely, frequent sets are extracted from distributed datasets and then are merged into a single frequent itemset. Moreover, after applying hierarchical clustering, indexing is measured for quality results [43]. Building analytical models of patients flow in hospitals is demonstrated in [77] using k-means clustering [82]. [77] emphasized on the data preparation phase as an essential step that affects the quality of solutions. Further, the size of the dataset is reported as main affecting factor to the solutions [77].

The design of the care model for patients for a given collection is reported in [22]. The approach in [22] identifies interested set of patients, then builds patients' care model (i.e., patterns of patients' care) and provides descriptions

of each pattern. Finally, these patterns are clustered to group similar patterns by exploiting agglomerative hierarchical clustering [81]. The work in [154] collects the real dataset of blood donors from Hacettepe University Hospitals' Computer Center. Exploiting Two-Step Cluster method [142] for clustering, Classification and Regression Tree (CART) [20] method for classification, Testik et al. [154] discovers arrival patterns of the blood donors.

A graph *b-coloring* clustering technique is reported in [47], which is compared with agglomerative hierarchical clustering [81] and an existing system in the french healthcare system. The approach aimed to find the hospital stays of patients. A probabilistic clustering model is proposed in [99] for the high-dimensional, temporal sparse, and uncertain electronic healthcare data. The model exploited *empirical prior distribution* for dealing the sparsity issues of the ten years data, collected from pediatric intensive care unit (PICU) at Children's Hospital Los Angeles.

2.3 Extraction of medical pathways

The medical pathways (also called clinical pathways) actually done by patients are extracted by using the well established sequential pattern mining technique. Sequential pattern mining [4] aims at identifying and extracting frequent sequences of events from the data collection. In addition, it plays an essential role in healthcare data for identifying hidden and interesting patterns for a given pathology.

The PhD research activity presents identification, detection, and evaluation of hidden patterns from raw healthcare data called exam-log data, which is an electronic storage of events of the physical diagnostic medical examinations (i.e. medical tests or services) with their corresponding timestamps. More specifically, the exam-log data has been analyzed to extract interesting medical pathways against the standard medical pathways (i.e., care guidelines). The research activity rebuilds the actual patients' treatment procedures (i.e. medical or clinical pathways) from an operational raw medical data and allows to detect the following information:

1. Set of examinations frequently done together
2. Sequences of set of examinations frequently done
3. Set of sequences frequently followed

An examination (hereafter "exam") is a physical diagnostic test performed by a patient for a certain pathology. A set of examinations (hereafter "exam set") is a collection of exams, which reports the exams done together by patients. A sequence of set of examinations (hereafter "exam sequence") actually reveals the temporal relation between the exam sets followed by patients. Formally exam set and exam sequence are described in the Definitions 2.3.1 and 2.3.2 respectively.

Definition 2.3.1 Exam Set. *Let $E = \{e_1, e_2, e_3,..., e_n\}$ be the set of exams done by patients in a given healthcare data. An exam set $\mathcal{S}$ is a group of exams such that $\mathcal{S} \in E$ and $1 \leq \mathcal{S} \leq |E|$.*

Definition 2.3.2 Exam sequence. *Let $E = \{e_1, e_2, e_3,..., e_n\}$ be the set of exams and $T = \{t_1, t_2, t_3, ..., t_n\}$ be the corresponding timestamps of the set of exams for patients. An exam sequence $\mathcal{S} = \{\{s_1\}\{s_2\}\{s_3\},...,\{s_n\}\}$ if $t_1 < t_2 < t_3 < ... < t_n$ is an ordered temporal relationship among set of exams such that $\mathcal{S} \in E$, where set of exams represent the order of exams in which these are diagnosed.*

Definition 2.3.3 Frequent exam sequence. *Let $\mathcal{S}$ be the exam sequence in a database $\mathcal{D}$. A frequent exam sequence $\mathcal{S}'$ is an exam sequence if its frequency is higher than the given minimum support threshold, where $\mathcal{S} \sqsubseteq \mathcal{S}'$.*

A sequence length is the number of exam sets present in an exam sequence performed by a patient for a given pathology. For example, let $S = \{\{e_1, e_2\}\{e_3\}\}$ be an exam sequence performed by a certain patient, which comprises of two exam sets $\{e_1, e_2\}$ and $\{e_3\}$. Further, exam set $\{e_1, e_2\}$ is done before $\{e_3\}$, while sequence length of the exam sequence S is 2, since two exam sets are present in S. Furthermore, a frequent exam sequence (see Definition 2.3.3) has the support frequency equal to or higher than a given value and the support of an exam sequence (or exam set) is the frequency (i.e., number of occurrences) of the exam sequence (or exam set) in a given database.

The extracted knowledge highlights the medical pathways typically adopted by patients for a specific disease as well as the deviation amongst pathways. In the following description of frequent patterns and frequent closed patterns is presented.

Table 2.1: Sequence Database $\mathcal{D}$

Ids	*Sequences*
1	A B C C A
2	A B C
3	B C D
4	A C D

2.3.1 Frequent patterns

Frequent pattern mining is one of the robust aspects of data mining for frequent data analysis. A pattern is a data behaviour, an arrangement or a form of data that might be of interest. A frequent pattern has support frequency equal to or higher than a given threshold. Several known algorithms have been exploited for finding frequent patterns/sequences such as Apriori [3] and PrefixSpan [119]. Frequent pattern mining is an expensive technique in terms of storage and computational power due to large data collections and thus, produce a large number of frequent sequences (FS) (or patterns). To prevent the cost of storage and computational power, several algorithms have been proposed to detect frequent closed sequences (FCS), described in the subsequent sections.

2.3.2 Frequent closed patterns

The frequent closed pattern (or sequence) represents compact form of the frequent sequences (i.e., frequent exam sequence). In other words, all the frequent sequences (i.e., exam sequences) are contained in frequent closed sequences (see Definition 2.3.4). The BIDE algorithm [163] is an example of the frequent closed sequence/pattern mining algorithms.

Definition 2.3.4 Frequent closed exam sequence. *Let $\mathcal{S}$ be a frequent exam sequence. A frequent sequence $\mathcal{S}'$ is a frequent closed exam sequence if there exists no proper super-sequences of $\mathcal{S}$.*

For example, consider the sequence database $\mathcal{D}$ reported in Table 2.1. The frequent sequences in the format *sequence:support* (i.e. C:4 indicate that the sequence containing item C has support 4) are the following:

A:3, B:3, C:4, D:2, AB:2, AC:3, ABC:2, BC:3, CD:2

The closed frequent sequences are the following:

C:4, AC:3, BC:3, CD:2, ABC:2

It is worth mentioning that items `A`, `B`, `D` and `AB` are not frequent closed items because their super-sequences `AC`, `BC`, `CD` and `ABC` have their same support.

The compression factor (CF) evaluates the compactness of frequent closed sequences (FCS) instead of considering all frequent sequences (FS). The CF can be defined as:

$$CF = (1 - \frac{\#FCS}{\#FS})\% \tag{2.1}$$

where $\#FCS$ is the number of all the frequent closed sequences and $\#FS$ is total number of frequent sequences at a given threshold value (i.e., *minsup*). For example, let $\mathcal{D}$ be a sequence database containing the following two sequences:

$$s_1 = \{e_1\}\{e_2, e_3\}\{e_2\} \text{ and } s_2 = \{e_2, e_3\}\{e_4\}$$

Consider *minsup = 2* (i.e., *100%*). Then 3 frequent sequences are generated i.e., $\{e_2\}$, $\{e_3\}$ and $\{e_2, e_3\}$, while the frequent closed sequence is only $\{e_2, e_3\}$. Sequences $\{e_2\}$, $\{e_3\}$ are not closed sequences since they are contained in sequence $\{e_2, e_3\}$.

When considering *minsup = 1*. Frequent sequences 17 (seventeen), and frequent closed sequences 3 (three) are extracted. The compression factor *CF* = 82% is achieved. Hence, frequent closed sequences significantly reduce the size of the solution set.

The BIDE algorithm

The BI-Directioanl Extension (BIDE) algorithm [163] is an efficient algorithm for mining frequent closed sequences. A sequence S is called closed sequence if there exists no proper super-sequences of S that has the same support as that of S [163]. The BIDE reported in *Algorithm 1* uses depth-first-search technique to traverse the sequence tree. More precisely, it checks the closed sequences by applying BI-Directional Extension closure checking without candidate maintenance. This closure checks forward-extension events and backward-extension events to declare a *prefix sequence* as frequent closed sequence. The prefix sequence is defined in the Definition 2.3.5.

Definition 2.3.5 Prefix sequence *Let $\mathcal{S} = \{e_1, e_2, e_3,..., e_n\}$ be a sequence. A subsequence $\mathcal{S}$ is prefix i-sequence e_1 of the $\mathcal{S}$ from the beginning of $\mathcal{S}$ to first appearance of item e_1 in $\mathcal{S}$.*

Algorithm 1 `BIDE` algorithm [163]

– *BIDE($\mathcal{S}$, minsup)*

Require: *A sequence database $\mathcal{S}$ and minsup*

Ensure: *Closed frequent sequences $(S_k) \geq minsup$*

1: $S_k = \phi$
2: *Scan $\mathcal{S}$ once to find frequent $1-$sequences F*
3: **for** *each frequent sequence S_p in F* **do**
4: *Build pseudo projected database S' of S_p $\{S_p$ is prefix sequence$\}$*
5: **end for**
6: **for** *each frequent sequence S_p in F* **do**
7: *Use BackScan pruning method for checking if S_p can be pruned $\{$BackScan uses ScanSkip to speed up the process$\}$*
8: **if** $!Pruned$ **then**
9: *Compute backward $-$ extension $-$ item BEI of S_p*
10: *bide$(S', S_p, BEI, minsup)$ $\{$call bide procedure$\}$*
11: **end if**$\{if$ *it can not be pruned*$\}$
12: **end for**
13: *Return S_k*

– *bide(S', S_p, BEI, minsup)*

Require: *a projected sequence database S', prefix sequence S_p, minsup and back $-$ extension $-$ items BEI*

Ensure: *set of closed frequent sequences S_k*

1: *find local frequent items FI in S'*
2: *Compute forward $-$ extension $-$ item FEI of S_p*
3: **if** $BEI + FEI = 0$ **then**
4: $S_k = S_k \bigcup S_p$
5: **end if**
6: **for** *each S'_p in FI* **do**
7: *Build pseudo projected datatbase S'' of S'_p*
8: **end for**
9: **for** *each S'_p in FI* **do**
10: *Use BackScan pruning method for checking if S'_p can be pruned*
11: **if** $!Pruned$ **then**
12: *Compute backward $-$ extension $-$ item BEI of S'_p*
13: *bide$(S'', S'_p, BEI, minsup)$ $\{$bide procedure recursively calls itself$\}$*
14: **end if**
15: **end for**
16: *Return S_k*

Definition 2.3.6 Projected sequence. *Let $\mathcal{S} = \{e_1, e_2, e_3,..., e_n\}$ be a sequence and prefix i-sequence e_1 be a prefix sequence of $\mathcal{S}$. A subsequence $\mathcal{S}'$ is a projected sequence, when first instance of prefix i-sequence e_1 is removed from $\mathcal{S}$.*

For example, consider sequence ABCCA. The *prefix sequence*-BC in sequence ABCCA is ABC. It should be noted that the first appearance of itemset BC is contained in its prefix sequence. The remaining part of the sequence ABCCA is CA, It is called projected sequence (see Definition 2.3.6) with respect to *prefix sequence*-BC. The complete set of projected sequences in a given sequence database $\mathcal{D}$ with respect to prefix sequence e_1, e_2, e_3,..., e_n is referred to as *projected database* in $\mathcal{D}$ with respect to the prefix sequence e_1, e_2, e_3,..., e_n.

For example, consider the sequence database $\mathcal{D}$ shown in Table 2.1. The projected database of *prefix sequence*-BC is {CA, ϕ, D, ϕ}, where ϕ indicates empty set. The forward-extension event checking generates locally frequent items with support of a *prefix sequence* since no candidates are generated. While backward-extension event checking is performed to handling the new sequence, which can be absorbed into already closed sequence.

The BIDE algorithm is very efficient because it saves time and tells about closed frequent sequences in which all the frequent sequences are already contained. Like PrefixSpan algorithm [119], it also works on projected database sequences and does not generate candidates thus memory space is saved. Moreover, BIDE also does not keep history of all frequent sequences to find closed ones and is a time efficient algorithm. The algorithm consumes time during forward-extension and backward-extension phases and possesses linear scalability property in terms of the number of sequences in a database.

2.3.3 Experimental results

The extracted medical pathways from the real datasets are presented in this section, particularly, frequent closed exam sets (hereafter "frequent exam sets") and frequent closed exam sequences (hereafter "frequent exam sequences"). Initially, the raw medical data has been prepared for the knowledge extraction process. In the following, details about data preparation are reported.

2.3.3.1 Data preparation

The main terminologies concerning data, such as data set and data attribute are described in the following.
Data. Data can be numbers, characters, images, or other method of recording that could be assessed by human. In computer terminology, it is collection of facts and figures, and can be processed, interpreted, stored, and/or transmitted on some digital channel.
Data set. Data set is a collection of data objects, and a data object could be a record, point, vector, pattern, event, observation and entity.
Data attribute. An attribute is a property of an object and attributes of different objects may differ. For example, "person" is a data object, whose attributes can be "height", "hair color" and "age". Thus, different persons can have different attributes.

The data preparation is one of the first essential steps to transform data into a suitable format, which can be further processed. The data is processed to remove unnecessary attributes and missing values. Then, the cleaned data can be transformed into specific format suitable for the post-processing operations. The complete procedure of data preparation is illustrated in Fig. 2.2.

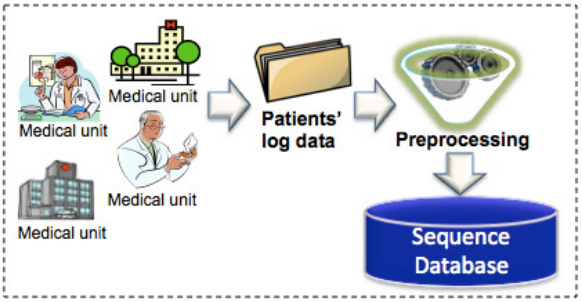


Figure 2.2: Data collection and preparation

The real raw healthcare data (i.e., exam-log data) were collected from medical units of Local Sanitary Agency of the Asti province - Italy. Then the collected data were integrated into a single common structure. Table 2.2 reports few example records of exam-log data. Three different exam-log data (considered as reference case studies in the PhD research) comprises of three different types of pathologies: (i) Diabetic patients, (ii) Colon-cancer patients, and (iii) Pregnant women.

The provided exam log data has been preprocessed to remove noisy and irrelevant data attributes. All irrelevant data attributes such as medical

Table 2.2: Patients' exam-log data

Patient ID	*Date*	*Exam*	...
1	10/01/07	Capillary blood	...
1	10/01/07	Glucose	...
2	25/05/07	Eye examination	...
3	15/03/07	Glucose	...
3	05/04/07	Venous blood	...
3	05/04/07	Urine Test	...
...	...	...	...

Table 2.3: Sequence database

Patient ID	*Exam sequence*
1	{Capillary blood, Glucose}
2	{Eye examination}
3	{Glucose} {Venous blood, Urine Test}
...	...

branch codes and description about the diagnostic exams, patients' addresses and other information have been removed and only attributes *patient id, exam date, exam name* are selected. Then, the cleaned and integrated data have been transformed into a sequence database (see Definition 2.3.7). The data transformation is reported in the following.

Definition 2.3.7 Sequence Database. *Let* $\mathcal{P} = \{p_1, p_2, p_3, ..., p_n\}$ *be the set of patient-identifiers and* $\mathcal{E} = \{e_1, e_2, e_3, ..., e_n\}$ *a temporal list of exam sets done by the patients in* $\mathcal{P}$*. A sequence database* $\mathcal{D}$ *is a collection of tuples, where each tuple is a set of pairs* (p_s, e_i)*,* $p_s \in \mathcal{P}$*,* $e_i \in \mathcal{E}$*.*

Data transformation The data is converted into a specific format suitable for subsequent operations. For example, in the medical exam log data, the name of the exams and their timestamps along with patients identifiers are needed for analyzing treatment procedures (i.e. medical pathways). Table 2.3 represents the outcome of the data transformation process. The exams of each patient are represented in such a way that exams done in same dates come together within a subgroup delimited by comma, where exams done on different dates are delimited by curly braces. For instance, Table 2.2 indicates that Capillary blood and Glucose exams are done by the patient id-1 at the same date. Similarly, the exams performed at different dates are

Table 2.4: Characteristics of exam-log data

Sr. No.	*Exam-log Name*	*Exam-log records*	*No. of patients*	*No. of distinct exams*	*Avg. exam length per patient*
1	Diabetic	95788	6380	159	15
2	Colon-cancer	2071	157	123	13
3	Pregnancy	29679	905	327	32

separated with curly braces (see patient id-3 in Table 2.3).

Data characteristics The three raw exam-log data considered in this study are transformed into the corresponding sequence databases as input to knowledge discovery process. Table 2.4 reports the characteristics of each exam-log data. The diabetic dataset comprises of 95788 log records of 6380 patients, the colon-cancer dataset has 2071 logging exams records of 157 patients and the pregnancy dataset exam-log data contains 29679 log-records of 905 patients. The numbers of distinct exams show the presence of different exams in the exam-log data, where as average exam length is the mean of number of exams done by a patient. The diabetic dataset contains minimum exam number 1 and maximum 154, while minimum 1, maximum 92 for the colon-cancer dataset, and 1 as minimum, 172 as maximum exams are recorded for the pregnancy dataset.

Data segmentation Data segmentation can be defined as a process or an activity of partitioning a given data set into small segments (chunks or groups) based on a specific criterion. The segmentation process needs some information about the data objects for portioning. For example, In the sequence database of pregnancy dataset, the amniocentesis exam is a crucial exam during pregnancy period. The women, who did amniocentesis exam differ from the rest of women in the database. These differences lead towards insight knowledge about the actual patterns. Therefore, sequence databases may be segmented for investigation of differences among various patients of a pathology.

The extraction of medical pathways from each of the three real sequence databases of different pathologies are reported in the subsequent sections.

2.3.3.2 Case study - Diabetic sequence database

The extracted medical pathways from the diabetic sequence database (see Section 2.3.3.1) are of two types: (i) frequent exam sets performed together

in the same date and (ii) frequent sequences of exam sets. The adopted approach is shown in Fig. 2.3 that includes additional information as shown in the block Domain constraints. Domain constraints help to focus on specific data analysis and following domain constraints:

Target exam set. The most expensive or crucial exam sets are considered. The target exam sets are selected from exam log data during the data preparation phase (see Section 2.3.3.1). Only those patients who have been diagnosed with target exams are considered. This approach is particularly essential while dealing with infrequent critical exams for a given pathology.

Medical pathways length. This constraint helps to focus on the analysis of patient specific clinical history (e.g., exams done multiple times in a year). This constraint is also inserted in the data preparation phase to discard the patients whose clinical history does not meet the constraint.

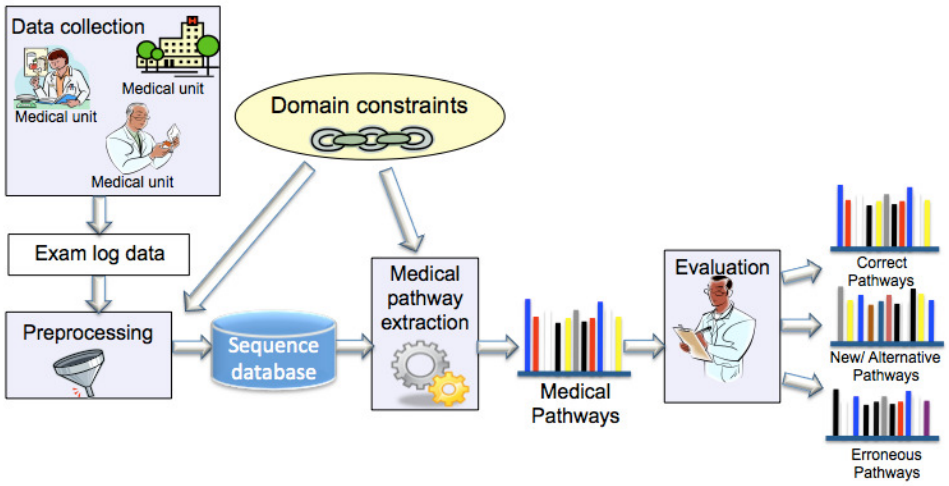


Figure 2.3: Medical pathways extraction process (diabetic dataset)

Furthermore, the extracted medical pathways have been evaluated with the support of a medical expert in accordance with medical domain knowledge and available medical guidelines. During the evaluation, the following scenarios have been detected:

Correct pathways. The extracted medical pathways are coherent with the medical knowledge. Thus, the treatment process followed by such patients is correct.

New or alternative pathways. The extracted medical pathway is not available in the medical domain knowledge. For example, the available medical guidelines do not cover rare and specific cases of a given disease. This scenario allows to identify common medical pathways followed by patients,

who could be exploited in the assessment of new guidelines.
Erroneous pathways. The extracted medical pathway is erroneous because it is not found coherent with medical domain knowledge. For example, it comprises of different or additional exams, or some exams are missing. These pathways could be for instance because of incorrect data collection procedures.

The detailed analysis of the extracted medical pathways for the diabetic sequence database is described in the following. In particular, exam frequencies, frequent exam sets, frequent exam sequences and medical pathways derived by injecting the *target exam set* constraint.

Exam frequencies

Though diabetic sequence database contains the medical exam-log data of one single year, a wide spectrum of clinical treatments is covered. The most frequent exams reflect the standard and routinely check-ups exams of diabetic patients for monitoring sugar concentration present in the blood. For example, glucose level (84.76%), venous blood (79.25%), capillary blood sample (75.03%), and urine test (74.87%). Serious diabetes complications are also detected with crucial exams having lower frequencies. The total cholesterol level (35.96%) and the triglycerides level (35.69%) exams detect the cardiovascular complications. The exams concerning liver disease alanine aminotransferase enzyme (30.14%) and aspartate aminotransferase enzymes (29.51%) are some examples of the complex cases. In the diabetic treatment, exams monitoring the eye status are also available to diagnose possible problems on the eye retina (retinopathy), e.g., the examination of fundus oculi (27.24%) and laser photocoagulation (2.24%). The later exam, being more specific for retina repairing, has a significantly lower frequency.

Frequent exam sets

The frequent exam set represents the exams frequently done together in the same day by the patients. The extracted exam sets are found consistent to medical domain knowledge for the diabetic treatment, but few anomalies are also detected. The glucose level is usually measured in association with atleast one of the three exams venous blood, capillary blood and urine test. For example, exam sets {glucose level, urine test} (74.86%), {glucose level, capillary blood} (74.40%), {glucose level, venous blood} (70.99%) have been extracted having higher frequencies. Besides, 5.56% of the patients have been

found, whose exam sets have atleast once glucose level exam not associated with any of the three exams. Such sequences clearly reflect an error condition, because the evaluation of glucose level exam without any of the three exams is not possible. These errors may have been present in the dataset because of incorrect data entry process, where incomplete data have been stored.

Frequent exam sequences

The exam sequences are found coherent with diabetic treatment. For example, glucose measure for monitoring disease status throughout the year are repeatedly performed. The most frequent sequences for the diabetic disease describe this behaviour, e.g., glucose level is repeated two times (58.20%) of patients, three times (31.83%) and four times (14.78%) during the considered year of the dataset.

Pathways including target exams

The extraction of the medical pathways related to the specific exams from the diabetic dataset have been carried out by selecting a subset of the dataset in data preparation step. The subset comprises of patients, who did atleast one of the target exams. For example, the damage of eye retina (retinopathy) is a serious diabetic disease degeneration. To repair retina lacerations, the retinal photocoagulation therapy is used. The patients affected of retinopathy usually require multiple therapy treatments. The subset of 143 patients (i.e., 2.24% of the total patients) including retinal photocoagulation therapy have been selected for the analysis of the patients who did atleast once retinal photocoagulation therapy. The pathways provide the insight knowledge of the retinal disease patients, e.g., therapy is repeated two times (50.35% of the subset, but 1.13% of the total) or three times (25.17% of the subset, but 0.56% of the total).

2.3.3.3 Case study - Colon-cancer sequence database

Health problems require proper and accurate medication to cure deadly disease in time, such as cancer. Colon-cancer is one of the deadly diseases which may cause cancer-related death. However, early diagnosis may lead towards proper and complete cure. There is a large availability of diagnostic guidelines for the colon-cancer disease, for instance, ASCRS [7], Cancer

Care Ontario [114], Commissione Oncologica Regionale Health Care Guideline [125], Effective Health Care [24] [25], Institute for Clinical System Improvement [56], NCI [109], and RCS [111]. The possibility of a very general level agreement of all guidelines on the same sequence of exams could be contended. Medical guidelines, usually, try to provide a reliable and accurate diagnosis of the cancer presence and at the same time to avoid as much as possible the adoption of expensive or invasive procedures. The International Classification of Diseases, Clinical Modification (ICD IX-CM classification) [113] is used in assigning codes to diagnosis and procedures. In the diagnosis of colon-cancer, Colonoscopy is considered gold-standard examination. The screening guideline prescribes a general physical examination, followed by a colonoscopy, a closed biopsy, and a diagnostic ultrasound of abdomen. Metastasis at the lungs are instead sought through a chest radiograph and confirmed by a Computerized Axial Tomography (CAT) scan.

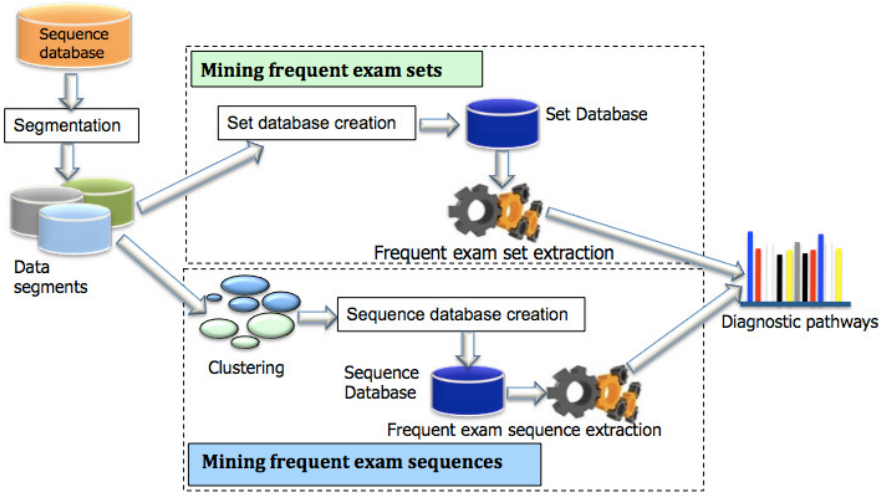


Figure 2.4: Medical pathways extraction process (colon-cancer dataset)

Guidelines generally do not suggest CAT of Abdomen as alternative method, instead these suggest the more invasive Colonoscopy [167]. Besides, the matter of fact is that the possibility of detection error by performing CAT is too high to suggest it. The methodology adopted for analyzing colon-cancer dataset is shown in Fig. 2.4. The sequence database created in the data preparation (see Section 2.3.3.1) has been segmented to group patients with similar behaviours. Then, frequent medical pathways (i.e. frequent exam sets and exam sequences) are extracted from each segment and compared to detect the differences in patients' behaviour (see Fig. 2.4).

Table 2.5: Characteristics of four segments (colon-cancer dataset)

Segment	*Diagnostic exam*	*Number of patients*
1	Colonoscopy or Closed Biopsy	32
2	CAT of Abdomen	37
3	X-Ray of Abdomen	50
4	None of the previous four diagnostic exams	38

The medical guidelines prescribe *only* colonoscopy and closed biopsy as diagnostic exams for colon-cancer. Alternative protocols occur when patients perform at least one different diagnostic exam, mainly CAT of Abdomen or X-Ray of Abdomen. The four main segments have been identified from colon-cancer sequence database, whose characteristics are summarized in Table 2.5. The patients of Segment_1 show *only* colonoscopy or closed biopsy as diagnostic exams. They have been neither diagnosed CAT of Abdomen nor X-Ray of Abdomen. This segment represents patients, who followed only the prescribed exams in the guidelines. For this reason, it has been used as a reference case for the medial pathway analysis. Likewise, Segment_2 and Segment_3 represent patients who have been diagnosed with least one diagnostic exam not prescribed in the guidelines.

For example, Segment_2 contains patients, who did at least once CAT of Abdomen exam along with other diagnostic exams, and those who did at least once X-Ray of Abdomen exam are in Segment_3. The patients, who did not follow the above four diagnostic exams, even if they were affected by the disease, are placed into Segment_4. Moreover, the Segment_2 and Segment_3 are not mutually exclusive i.e., patients, who did both CAT of Abdomen and X-Ray of Abdomen exams belong to both segments. However, both segments do not overlap with the Segment_1 and Segment_4. The main blocks the approach are described in the following.

Mining frequent exam sets The analysis has been carried out in each segment for the identification of the frequent exam sets. In Fig. 2.4, the upper part presents the main steps. Firstly, the exam log data transformed into a sequence database is segmented. Then, data segments are represented as a set database reporting the different exams done by each patient. All the irrelevant information about the exams have been omitted for the analysis such as when exams have been diagnosed (i.e., timestamps) and redundancy (i.e., repetition of the exams). Then, the mining process is applied to extract frequent exam sets on each segment separately.

Table 2.6: Statistics of four segments (colon-cancer dataset)

	Avg. num. of cluster per patient	Avg. num. of exams per cluster	Avg. time dimension per cluster
Segment_1	3.0	2.7	3.5
Segment_2	3.0	5.7	7.3
Segment_3	2.6	7.0	7.0
Segment_4	1.9	3.6	6.3

Mining frequent exam sequences The temporal relationships among exams, i.e., which exams frequently precede or follow other exams, have been investigated by considering the analysis of frequent exam sequences. The temporal order of two consecutive exams may not be always relevant. For instance, when two consecutive exams have the difference of few days, the order of the exams could be because of scheduling reasons instead of constraints for the prescription. Therefore, the adopted approach applies a clustering algorithm to group together the exams diagnosed in a '*close*' period of time. Each cluster represent a set of exams. However, the order of the exams within the cluster has not considered. Further, sequence database is built from clusters and ultimately analyzed to extract frequent exam sequences, as reported in the lower part of Fig. 2.4.

Clustering of exams The DBSCAN algorithm [50] (see Section 2.5.2) has been applied to group together exams in a '*close*' time period. To group two consecutive exams into a single cluster, the notation of temporal distance between the exams should be defined. In the proposed approach, the maximum distance represents the maximum time interval between two consecutive exams. The euclidean distance (see Section 2.5.3.1) is used to measure distance between two consecutive exams. The approach groups two consecutive exams if the maximum time interval between them is not greater than 12 days. This constraint caused an average cluster time dimension of around 7 days, that means that, on average, exams done within a week are considered in the same cluster.

The clustering results are reported in Table 2.6. Almost the same number of patients is grouped together in Segment_1, Segment_2, and Segment_3, whilst the number of patients is significantly decreased in Segment_4. The Segment_1 averagely has a lower number of exams per cluster in comparison with the other three segments. This behaviour of Segment_1 reveals that patients have quite homogeneous behaviour and they have been diagnosed exams in a very

short time interval. The patients in the other clusters instead have a quite diversified behaviour, since they did more exams at a larger time interval.

The analysis of the medical pathways extracted from the colon-cancer dataset is reported in the following.

Exam frequencies

Colonoscopy, Closed Biopsy, and Diagnostic Ultrasound of Abdomen exams specified in the care guidelines are the most frequent in Segment_1. While Electrocardiogram exam is often diagnosed before surgical operation. The rest of the exams in Fig. 2.5 helps to investigate side effects of colon cancer. The results of Segment_1 are consistent to medical guidelines for the colon cancer.

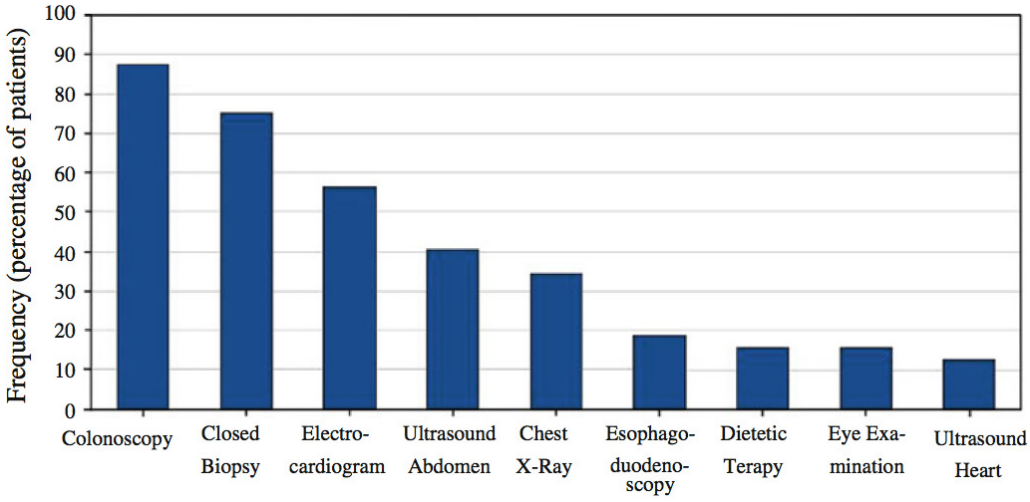


Figure 2.5: Exam frequencies in Segment_1 (colon-cancer dataset)

The behaviour of patients in Segment_2 and Segment_3 is more diversified than that of the Segment_1. For example, the frequency of exams in the two segments is higher than that of in Segment_1. Each patient has been diagnosed with a larger set of different exams (averagely 8.7 in Segment_2 and 7.9 in Segment_3) and a large number of different exam sets is present in both segments (178 sets in Segment_2 and 189 in Segment_3, that is almost three times the number of exam sets in Segment_1). Precisely, the most frequent exams in both segments are almost the same and with slightly differences in their frequencies. For instance, Routine Chest X-Ray (49.4% in Segment_2 and 58% in Segment_3), Diagnostic Ultrasound of Urinary System (18.5% in

$Segment_2$ and 11% in $Segment_3$), and Diagnostic Ultrasound of Digestive System (29.3% in $Segment_2$ and 20.9% in $Segment_3$). The later two exams have been diagnosed by single patient in $Segment_1$. To detect possible metastasis typically two exams CAT Scan of Thorax (43.2% in $Segment_2$ and 20% in $Segment_3$) and CAT Scan of Head (13.5% in $Segment_2$ and 10% in $Segment_3$) are found most frequent ones.

Since $Segment_4$ comprises of patients who have been diagnosed none of the four exams (i.e., Colonoscopy, Closed Biopsy, CAT of Abdomen and X-Ray of Abdomen) recommended in medical guidelines, therefore, no medical pathways have been extracted. Moreover, most of the patients were diagnosed one or two exams. These exams may reflect the alternative diagnosis for the colon cancer. A possible reason for lack of diagnostic exams may be the data entry errors or the patients may have been privately diagnosed. Another possible explanation may be that some patients have been diagnosed during surgical operation.

Frequent exam sets

The analysis of frequent exam sets highlight the homogeneous behaviour of the patients in $Segment_1$, few different exams i.e., on average 4.8 exams are the most frequent for each patient. Moreover, the number of different exams are quite low in this segment (about 58). Only few patients did larger set of exams having different exams. The larger exam sets may have occurred due to complication of disease or some additional exams for other pathologies. For example, 25% patients have done more than 6 different exams, and 6.3% did more than 10. One patient has been found having more than 10 exams, including Bronchoscopy and Bronchial Biopsy. These exams probably may be due to another pathology.

In $Segment_1$, 62.5% of patients comply with medical guidelines by doing both Colonoscopy and Closed Biopsy exams. While 37.5% have been diagnosed having only one of the two exams i.e., 25% did only Colonoscopy and 12.5% did only Closed Biopsy. This may be an erroneous condition, since Colonoscopy should always come before Closed Biopsy for the provision of the tissue sample needed for later exam. This error condition may be due to incorrect data entry, or some exams were done privately. Thus only partial entry has been recorded. The patients, who strictly comply with the medical guidelines are 18.8%, and they were diagnosed Colonoscopy, Closed Biopsy, and Diagnostic Ultrasound of Abdomen. In both segments $Segment_2$ and $Segment_3$, the large number of different exams are done. This behaviour

Table 2.7: Exam sequences in Segment_1 (colon-cancer dataset)

Exam sequence	*Frequency (%)*
{Colonoscopy, Closed biopsy}	62.5
{Colonoscopy, Closed biopsy}, {Electrocardiogram}	28.1
{Diagnostic Ultrasound of Abdomen}, {Colonoscopy}	25.0
{Colonoscopy}, {Routine Chest X-Ray}	25.0
{Electrocardiogram}, {Routine Chest X-Ray}	21.9
{Colonoscopy, Electrocardiogram}	18.8
{Colonoscopy, Diagnostic Ultrasound of Abdomen}	15.6
{Electrocardiogram, Routine Chest X-Ray}	12.5
{Electrocardiogram}, {Colonoscopy}, {Electrocardiogram}	12.5

leads towards a significant increase in costs, for instance, CAT Scan of Abdomen is found together either with Colonoscopy (46% patients) or X-Ray of Abdomen (62.2% patients) in Segment_2. X-Ray of Abdomen exam found together with CAT Scan of Abdomen 46% and Colonoscopy 26% in Segment_3. Whilst medical pathways did not emerge in Segment_4.

Frequent exam sequences

The most frequent exam sequences extracted in Segment_1 and Segment_2 are reported in Tables 2.7 and 2.8 respectively. The clustered exam sets are delimited by two brace brackets and are diagnosed within a limited time interval (i.e., 12 days). For example, referring to Table 2.7, both exams Colonoscopy and Closed biopsy are grouped together in the same cluster and 62.5% of patients followed this exam sequence. The two separate brace brackets represent two clusters of exams, e.g., 28.1% patients did the exams Colonoscopy and Closed biopsy both in the same cluster and consecutive cluster has Electrocardiogram exam. In addition, some patients have performed repeatedly same exams. For instance, Electrocardiogram is done twice by 12.5% of patients in the same cluster. The repetition of exams may be due to the emergency hospitalization.

CAT Scan of Abdomen is diagnosed often with either Electrocardiogram or Routine Chest X-Ray in Segment_2. Likewise, 43.2% of patients did Colonoscopy, additionally they did CAT Scan of Abdomen in the same cluster. The repetition of the exams is more frequently done in Segment_2 in comparison of Segment_1, for instance, {Routine Chest X-Ray, Routine Chest X-Ray} 59.5%, {X-Ray of Abdomen, X-Ray of Abdomen} 45.9% and {Electrocardiogram, Electrocardiogram} 37.8% of patients. However, the exam sequences {CAT Scan of Abdomen}, {X-Ray of Abdomen} 21.6% and

Table 2.8: Exam sequences in $Segment_2$ (colon-cancer dataset)

Exam sequence	*Frequency (%)*
{CAT Scan of Abdomen, Electrocardiogram}	70.3
{CAT Scan of Abdomen, Routine Chest X-Ray}	67.6
{Electrocardiogram, Routine Chest X-Ray}	64.9
{Routine Chest X-Ray, Routine Chest X-Ray}	59.5
{Routine Chest X-Ray, X-Ray of Abdomen}	56.8
{X-Ray of Abdomen, X-Ray of Abdomen}	45.9
{CAT Scan of Abdomen, CAT Scan of Thorax}	43.2
{CAT Scan of Abdomen, Colonoscopy}	43.2
{Electrocardiogram, Electrocardiogram}	37.8
{CAT Scan of Abdomen}, {X-Ray of Abdomen}	21.6
{X-Ray of Abdomen}, {CAT Scan of Abdomen}	16.2
{CAT Scan of Abdomen, CAT Scan of Thorax}, {Routine Chest X-Ray}	10.8
{CAT Scan of Abdomen}, {Routine Chest X-Ray}, {Routine Chest X-Ray}	13.5

{X-Ray of Abdomen}, {CAT Scan of Abdomen} 16.2% of patients reflect that there is no any clear evidence for the ordering of both exams. CAT Scan of Abdomen follows X-Ray of Abdomen by 21.6% of patients, while the reverse order appeared 16.2% of patients. The results of $Segment_3$ are similar to $Segment_2$, Routine Chest X-Ray is usually repeated. This behaviour is not only appeared in the same cluster but also in longer time intervals (i.e., in consecutive clusters). For example, {Routine Chest X-Ray} {Routine Chest X-Ray} is diagnosed by 28% of patients.

2.3.3.4 Case study - Pregnancy sequence database

Determination of fetus conditions, such as whether it comprises of certain abnormalities including established hereditary or instinctive genetic disorders, is monitored by testing the fetus before birth. Generally prenatal diagnostic testing involves such sort of testing, some of the tests, for instance, blood tests and Ultrasonography, are part of routinely prenatal care [57] [110].

Prenatal diagnostic exams are often distributed on weekly basis, for example, within first thirteen (13) weeks Complete Blood Count (CBC), Toxoplasmosis antibody, Rubella Virus Antibodies, Virus Immune Deficiency Antibodies (HIV), Urinalysis (Urine microscopic exam), Obstetric Ultrasound (Ultrasonography), concentration of Glucose level in the blood, and Indirect Coombs Test or Anti-erythrocyte antibody detection are performed. The duration between 19^{th} and 23^{rd} week needs repetition of Ultrasonography and Urinalysis examinations. The Urinanalysis and concentration of Glucose level in the blood are repeated in between 24^{th} and 27^{th} week of the

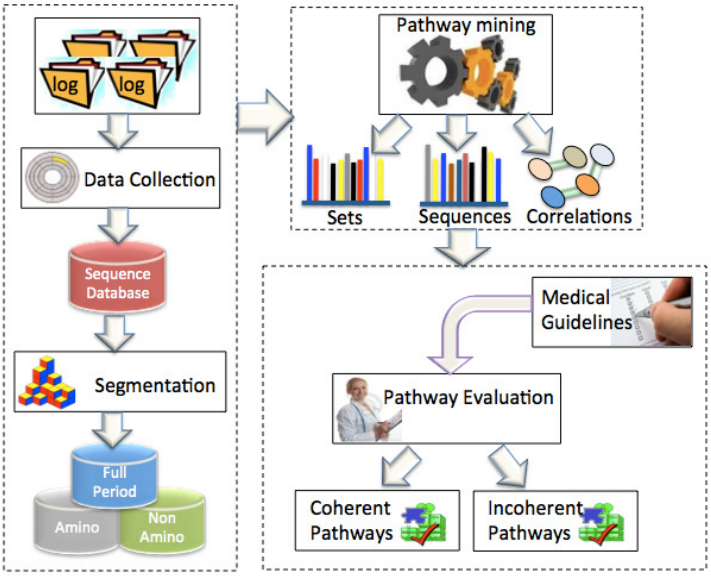


Figure 2.6: Medical pathways extraction process (pregnancy dataset)

pregnancy period. Whilst in the period of 28^{th} and 32^{nd} week, Urinalysis, CBC, and Ultrasonography are performed again. Then, between the time period of 33^{rd} and 37^{th} week of pregnancy Hepatitis B antibody (HBV) and Hepatitis C antibody (HCV) tests are performed, and some tests are repeated including CBC, HIV and Urinanalysis. Finally, last weeks i.e., between 38^{th} and 40^{th}, Urinanalysis is repeated and in the 41^{st} week Ultrasonography and Cardiotocography tests are performed. [57]. The International Classification of Diseases, Clinical Modification (ICD IX-CM classification) [113] assigns codes to diagnosis and procedures. Table 2.9 reports timing of each prenatal exams as the medical guidelines for the pregnancy period.

The adopted approach for extracting medical pathways from pregnancy dataset is depicted in Fig. 2.6. The approach mainly comprises of three phases. Firstly, the collected data collection is transformed into sequence database and then, it is segmented to analyze the small group of patients separately. The considered dataset has 29679 logging records (see details in Table 2.4). All the exams performed by the 905 women who gave birth to a child are recorded in between July and December 2007. Secondly, pathways mining phase does the actual extraction of the medical pathways. Finally, the obtained results are evaluated based on available medical guidelines in the pathway evaluation phase.

The transformed sequence database has been segmented into three subsets

Table 2.9: Guidelines for pregnancy exams

Exam code	*Exam name (abbreviation)*	*week*
90.04.5	Alanine Aminotransferase (ALT)	1-13
90.09.2	Aspartate Aminotransferase (AST)	1-13
90.62.2	Complete Blood Count (CBC)	1-13
91.09.4	Toxoplasma Antibodies	1-13
91.22.4	Virus Immune Deficiency Antibodies (HIV)	1-13
91.26.4	Rubella Virus Antibody	1-13
90.27.1	Glucose	1-13
90.44.3	Urine Microscopic (Urinalysis)	1-13
90.49.3	Anti-Erythrocutes	1-13
88.78	Obstetric Ultrasound (Ultrasonography)	1-13
90.44.3	Urine Microscopic (Urinalysis)	14-18
90.44.3	Urine Microscopic (Urinalysis)	19-23
88.78	Obstetric Ultrasound (Ultrasonography)	19-23
90.27.1	Glucose	24-27
90.44.3	Urine Microscopic (Urinalysis)	24-27
90.62.2	Complete Blood Count (CBC)	28-32
90.22.3	Ferritin	28-32
90.44.3	Urine Microscopic (Urinalysis)	28-32
88.78	Obstetric Ultrasound (Ultrasonography)	28-32
91.18.5	Hepatitis B Virus (HBV)	33-37
91.19.5	Hepatitis C Virus (HCV)	33-37
90.62.2	Complete Blood Count (CBC)	33-37
90.44.3	Urine Microscopic (Urinalysis)	33-37
91.22.4	Virus Immune Deficiency Antibodies (HIV)	33-37
90.44.3	Urine Microscopic (Urinalysis)	38-40
88.78	Obstetric Ultrasound (Ultrasonography)	≥ 41
75.34.1	Cardiotocography (CTG)	≥ 41
90.94.2	Culture Urine Test (Uration)	≥ 41

of similar characteristics of patients namely $Segment_{Full-Period}$, $Segment_{Amnio}$ and $Segment_{Non-Amnio}$. The $Segment_{Full-Period}$ comprises of 455 patients that have almost full pregnancy period. However, not all patients full history (i.e., 9 months diagnostic records) is available, this may be due to some of patients had exams in a predicate structure. Therefore, for analyzing complete medical pathways of pregnant women, patients whose atleast 190 days (i.e., greater than 6 months) history is available are included in $Segment_{Full-Period}$.

Another group of patients is segmented based on specific diagnostic exam, which highlights abnormal conditions. In pregnancy, one of the crucial examination is *Amniocentesis*. This exam, being an invasive is used to determine genetic disorders about the unborn baby and this also diagnoses uterine infection. The examination of amniotic fluid that surrounds the unborn baby in the womb is carried out by Amniocentesis exam. Moreover, it is usually diagnosed after first 3 months of pregnancy period for women older than 35 years [57]. To analyze the medical pathways of women in critical conditions, the sequence database is segmented into two groups: (i) Patients with critical conditions and (ii) Patients without critical conditions.

The $Segment_{Amnio}$ contains 73 patients, who did the crucial examination (i.e., Amniocentesis) and the rest of the patients (i.e., 832 patients) from a total of 905 are grouped into $Segment_{Non-Amnio}$. Hence, the medical pathways have been extracted form both segments and their results are compared to get insight knowledge about the impact of critical conditions on the entire pregnancy period. Moreover, the analysis of $Segment_{Full-Period}$ have also been investigated to understand the complete prenatal care. In addition, the patients in $Segment_{Full-Period}$ are analyzed on the basis of trimester. A trimester comprises of 3 months time duration of pregnancy period, thus in 9 months there would be 3 trimesters: 1st Trimester, 2nd Trimester and 3rd Trimester. The initial 3 months are covered in 1st trimester, next 3 months in 2nd trimester and rest of the days are included in 3rd trimester. In the subsequent sections, the analysis of the extracted medical pathways is described.

Frequent exam sets

The frequent exam sets of $Segment_{Full-Period}$ trimester-wise are reported in Table 2.10, it also includes results as a whole complete period. Majority of the results are consistent to the guidelines reported in Table 2.9. The most frequent exams in 1st trimester, including Glucose, ALT, AST, HIV, Toxoplasma, and Ultrasonography, are those which are recommended in first

Table 2.10: Exam sets in $Segment_{Full-Period}$ (pregnancy dataset)

Exam sets	1^{st} *Trimester*	2^{nd} *Trimester*	3^{rd} *Trimester*	*All*
{Ultrasonography}	**89%**	59%	65%	89%
{Glucose}	**60%**	35%	50%	79%
{Toxoplasma}	**57%**	49%	47%	77%
{ALT}	**57%**	27%	50%	78%
{AST}	**56%**	27%	50%	77%
{ALT, AST}	**56%**	27%	50%	77%
{CBC, Urinalysis, Glucose}	**52%**	25%	37%	76%
{HIV}	**45%**	11%	31%	62%
{Rubella Virus Antibody}	**41%**	6%	3%	45%
{CBC, Urinalysis, HIV}	**40%**	8%	27%	59%
{Anti-Erythrocutes}	**35%**	11%	13%	45%
{Amniocentesis}	5%	**6%**	- - -	10%
{CBC}	66%	62%	**76%**	92%
{Urinalysis}	66%	72%	**73%**	91%
{HBV}	36%	13%	**49%**	67%
{Cardiotocography}	9%	8%	**45%**	47%
{Prothrombin time}	18%	12%	**45%**	50%
{Echocardiography}	8%	8%	**44%**	47%
{Partial thromboplastin time}	18%	12%	**44%**	50%
{Antithrombin III}	14%	14%	**40%**	44%
{HCV}	29%	11%	**39%**	55%
{Creatinine}	31%	16%	**34%**	48%
{Uration}	22%	14%	**34%**	42%
{Total Bilirubin}	25%	15%	**31%**	41%
{Ferritin}	13%	11%	**24%**	33%

weeks of the prenatal care. The exams Echocardiography, Uration, Uration, Total Biluribine, HBV, and HCV are the most frequent in 3rd trimester in accordance with medical guidelines. Th only Amniocentesis exam is found most frequent in 2nd trimester, this leads to the fact that Amniocentesis is usually performed in between 15th and 20th week of the pregnancy period. Thus, the most of the results are coherent with guidelines. The exam sets highlight the fact that these exams are performed together, for instance, ALT and AST.

There has also been some unexpected results in $Segment_{Full-Period}$, since some of the frequencies of the exams are lower than the expected ones, e.g., Rubella Virus Antibody, Cardiotocography, and Ultrasonography. The possible reason behind such behaviour could be these exams quickly and comfortably are performed privately. The patients may have preferred private examination of such exams to avoid long queues of the public health care centres. Another crucial exam Rubella Virus Antibody is found about 45%, that is the limited percentage of patients. The Rubella Virus Antibody exam

Table 2.11: Exam sets in $Segment_{Amnio}$ and $Segment_{Non-Amnio}$ (pregnancy dataset)

Exam sets	$Segment_{Amnio}$	$Segment_{Non-Amnio}$	*Difference*
{Ultrasonography}	100%	76%	24%
{CBC}	88%	77%	21%
{HBV}	68%	52%	16%
{CBC, Urinalysis, HBV}	63%	49%	14%
{HIV, Urinalysis, CBC}	56%	43%	13%
{HIV}	56%	45%	11%
{HBV, HCB}	53%	42%	11%
{HCV}	53%	43%	10%
{Urinalysis, Glucose, CBC}	68%	59%	9%
{Glucose}	73%	64%	9%
{Urinalysis}	85%	77%	8%
{CBC, ALT, AST}	67%	60%	7%
{ALT}	68%	61%	7%
{AST}	67%	61%	6%
{ALT, AST}	67%	61%	6%
{Toxoplasma}	63%	61%	2%

is very critical because, in case of infection found in mother, the baby may congenital rubella syndrome. This implies to a serious incurable illness. The possible reason of is limited frequency could be people did Rubella earlier, since they are already known about the antibodies. Thus, such behaviour of low frequency has been analyzed. Moreover, some of the exams are having higher frequency than that of the expected ones, such behaviour reveals the aspect of medical guidelines being obsolete or at least incomplete with respect the actual medical knowledge. Prenatal care is complex process, which is contingent on health conditions of the woman, therefore, medical experts (i.e., doctors) prescribe treatment in accordance with patient's actual conditions. The Creatinine (48%), Prothrombin Time (50%), Antithrombin III (44%) and Echocardiography (47%) are some examples of such exams, which are unavailable in guidelines, but appeared with higher frequency in the considered dataset. The guidelines prescribe Glucose level, ALT, AST once in the whole pregnancy period, but these exams are found with higher frequency in both the 1st and the 3rd trimesters.

The exam sets are generally more frequent in $Segment_{Amnio}$ with respect to $Segment_{Non-Amnio}$, e.g., atleast always 10% higher frequency is analyzed in $Segment_{Amnio}$ as compared to $Segment_{Non-Amnio}$ for the exams HIV, HBV and HCV. The frequent exam sets of both the segments are reported in Table 2.11. Although the exams ALT, AST and Toxoplasma are having almost similar frequencies in both the segments, yet $Segment_{Amnio}$ has slightly higher

Table 2.12: Exam sequences of three trimesters in $Segment_{Full-Period}$ (pregnancy dataset)

Exam sequences	*Frequency (%)*
{Ultrasonography}{Urinalysis}	73
{Ultrasonography}{CBC}	73
{Urinalysis}{Urinalysis}	64
{Ultrasonography}{Ultrasonography}	64
{CBC}{CBC}	60
{HIV}{Urinalysis}	38
{Glucose}{Glucose}	37
{Ultrasonography}{Urinalysis}{CBC}	44
{Ultrasonography}{Urinalysis}{ Urinalysis }	43
{Ultrasonography}{Urinalysis}{Urinalysis, CBC}	40
{Ultrasonography}{Ultrasonography}{CBC}	38
{Ultrasonography}{Ultrasonography}{ Urinalysis}	37
{Ultrasonography}{CBC}{CBC}	36
{Ultrasonography}{Urinalysis}{Ultrasonography}	36
{Ultrasonography}{Ultrasonography}{Ultrasonography}	35
{Ultrasonography}{Toxoplasma}{Urinalysis }	31
{Urinalysis}{Urinalysis}{Urinalysis}	27
{Ultrasonography}{Urinalysis}{Echocardiography}	26
{CBC}{CBC}{CBC}	24
{Ultrasonography}{Urinalysis}{Cardiotocography}	23
{Ulrasonography, HIV}{Urinalysis}{CBC}	21
{Toxoplasma, AST, ALT}{Urinalysis}{CBC, Urinalysis}	20

frequency. The diversified behaviour is due to critical conditions of patients in $Segment_{Amnio}$, since these patients are prescribed more and complete pregnancy exams. Furthermore, the Amniocentesis exam is done in women older than 35 years, hence prenatal care treatment is carried out carefully.

Frequent exam sequences

Table 2.12 reports the most frequent exam sequences across trimesters in $Segment_{Full-Period}$ (i.e., each exam set is extracted from a different trimester). The analysis highlights that the sequences are usually with quite lower frequency. This observation reveals the aspect that these exam sequences are not done by the majority of the women. However, the sequences with lower frequency are inconsistent to medical guidelines, the possible reason for this behaviour could be some of the patients may have done some exams privately. The sequence {Ultrasonography}{CBC} has 73% frequency indicates that 73% of patients did Ultrasonography in one trimester and CBC in the follow-

Table 2.13: Exam sequences in $Segment_{Amnio}$ and $Segment_{Non-Amnio}$ (pregnancy dataset)

Exam sequences	$Segment_{Amnio}$	$Segment_{Non-Amnio}$	*Difference*
{AST, ALT}{AST, ALT}	45%	- - -	45%
{Ultrasonography}{Ultrasonography}	82%	48%	34%
{CBC, Urinalysis} {CBC, Urinalysis}	73%	44%	29%
{CBC}{CBC}	78%	52%	26%
{CBC, Glucose}{CBC, Glucose}	51%	26%	25%
{Urinalysis}{Urinalysis}	78%	54%	24%
{Ultrasonography}{Ultrasonography}{Ultrasonography}	62%	29%	33%
{CBC}{CBC}{CBC}	59%	33%	26%
{Urinalysis}{Urinalysis}{Urinalysis}	51%	36%	15%
{Urinalysis}{Urinalysis}{Urinalysis}{Urinalysis}	- - -	22%	22%

ing trimester. The sequence {Ultrasonography}{Urinalysis}{CBC} shows that 44% of patients did Ultrasonography in the first trimester, Urinalysis in the second trimester and CBC in the third trimester. The exam sequences extracted from $Segment_{Amnio}$ and $Segment_{Non-Amnio}$ are described in Table 2.13.

2.4 Extraction of exam correlations

The examination correlation (hereafter "exam correlation") reveals the interdependency between the exams (i.e., diagnostic examinations performed by patients) and uncovers the associations between the exam sets for a given pathology. Thus exam correlation (i.e., association rule) helps to predict the series of exams needed for a given exam set. The exam correlations are extracted by means of association rule mining technique explained in the following sections.

2.4.1 Association rule mining

Association rule mining [2] is a popular method for discovering interesting relations between variables in databases. An association rule (see Definition 2.4.1) is usually represented as: If *Body* then *Head*, where *Body* (also called *antecedent*) and *Head* (also called *consequent*) are disjoint patterns. The association rule is usually represented as the following:

$$antecedent \Rightarrow consequent$$

Precisely, if *antecedent* happens then there are more chances that *consequent* may happen for a given strength of the association rule.

The implication between *antecedent* and *consequent* does not mean that if *antecedent* happens then *consequent* must also happen. The strength of the association rule is usually measured by the rule support and confidence. The support is the frequency of the rule i.e., number of occurrences of the rule in transactional dataset (see Definition 2.4.2). The confidence of a rule (see Definition 2.4.3) is the conditional probability in the transactional dataset of the *antecedent* given the *consequent*. The confidence measure sometimes misleads the correlation between the objects, especially when there is huge difference between frequencies of the rule *antecedent* and *consequent*. Some other measures for association rule evaluation are introduced such as *Lift* [153]. The lift of an association rule $A \Rightarrow B$ is described in Definition 2.4.4. If *lift(A,B)=1*, itemsets A and B are not correlated (i.e., they are statistically independent). The lift value below 1 indicates negative correlation, where as lift value above 1 describes a positive correlation between itemsets A and B. An association rule is called frequent if its support and confidence values are equal to or greater than the given minimum threshold values.

Definition 2.4.1 Association rule. *Let $\mathcal{F}$ be the set of all frequent items in a dataset $\mathcal{D}$. An association rule $\mathcal{R}$ is an implication of the form $A \Rightarrow B$, where $A,B \in \mathcal{F}$ are disjoint itemsets.*

Definition 2.4.2 Association rule support. *Let $A \Rightarrow B$ be an association rule in a transactional dataset $\mathcal{D}$. The rule support $Support(A \Rightarrow B)$ is the support of the itemset $A \bigcup B$ in $\mathcal{D}$.*

Definition 2.4.3 Association rule confidence. *Let $A \Rightarrow B$ be an association rule in a transactional dataset $\mathcal{D}$. The rule confidence $conf(A \Rightarrow B)$ is given by $\frac{Support(A \bigcup B)}{Support(A)}$, where Support(A$\bigcup$B) and Support(A) are support values in $\mathcal{D}$.*

Definition 2.4.4 Association rule lift. *Let $A \Rightarrow B$ be an association rule in a transactional dataset $\mathcal{D}$. The rule lift is given by $\frac{conf(A \Rightarrow B)}{Support(B)}$ or $\frac{Support(A \bigcup B)}{Support(A)Support(B)}$.*

2.4.1.1 The Apriori algorithm

The Apriori algorithm [3] applies *bottom-up* approach for finding frequent items in a given transactional database. The first pass of *Apriori* algorithm counts item occurrences (i.e., frequencies) to determine *length-1 itemsets*. An

itemset is a collection of one or more items in a given transactional database (see Definition 2.4.6). The formal definition of *item* is described in Definition 2.4.5. Items are extended, for instance, an item at a time to generate itemsets of *length-2, length-3* and so on; this extended step is termed as *candidate items*. The algorithm ends when no other extension of items is possible (i.e., no other frequent pattern is available). The *apriori* algorithm is presented in *Algorithm* 2.

Algorithm 2 Apriori algorithm [3]

Require: *A sequence database* $\mathcal{D}$ *and minsup*
Ensure: $frequent\ itemsets\ (F_k) \geq minsup$
1: $k = 1$
2: $Generate\ candidate\ itemsets\ (C_k)\ of\ length - 1$
3: **repeat**
4: $Extract\ length\ (k+1)\ candidate\ itemsets\ (C_k)\ from\ length - k\ frequent\ itemsets$
5: $Count\ frequency\ of\ each\ candidate\ item\ in\ C_k\ by\ scanning\ \mathcal{D}$
6: $Eliminate\ infrequent\ itemsets\ (i.e.,\ C_k \leq minsup)$
7: $Update\ F_k = F_k \bigcup C_k$
8: **until** *no new* $frequent\ itemsets\ are\ identified$
9: $Result\ F_k$

Apriori algorithm uses *breath-first* searching technique to find frequent items (see Definition 2.4.7) and *tree* data structure from the implementation point of view. For example, consider a database $\mathcal{D}$ containing itemsets shown in Table 2.1. The support (i.e., frequency) of an item i is the percentage of records in the database $\mathcal{D}$ that contains collection of items (i.e., $i_i \in \mathcal{D}$). An item `i` is called frequent if its support is equal to or greater than a specified support threshold called minimum support (i.e., *minsup*). The formal definition is described in Definition 2.4.7. Support count is frequency value of an item i contained in a given database $\mathcal{D}$.

Definition 2.4.5 Item. *Let* $\mathcal{I} = \{i_1, i_2, i_3, ..., i_n\}$ *be the set of all objects in a database* $\mathcal{D}$. *An instance* $\boldsymbol{i}$ *is an item such that* $i_i \in \mathcal{I} \in \mathcal{D}$.

Definition 2.4.6 Itemset. *Let* τ *be an enumeration of all items. An itemset* $X \subseteq \tau$ *is a set of items such that each item* i_i *may occur at most once in* X.

Definition 2.4.7 Frequent itemset. *Let* X *be the set of items such that* $X \subseteq \tau$, *where* τ *is an enumeration of all items. An itemset* $\mathcal{I}$ *is a frequent,*

Table 2.14: Frequent items

Items	*Support*
A	3
B	3
C	4
D	2

Table 2.15: Frequent itemsets

Items	*Support*
AB	2
BC	3
AC	3
CD	2

if the support of $\mathcal{I}$ is higher than a given minimum support threshold, where $\mathcal{I} \subseteq X \subseteq \tau$.

The Apriori algorithm initially counts support count for each item (i.e., *length-1 items*) separately as shown in Table 2.14. Next step is to generate candidate items of *length-2*. Lets consider *minsup* $\geqslant$ *2*, the *length-1* items and *length-2* itemsets are reported in Table 2.14 and Table 2.15 respectively. Since no other *length-2* itemsets satisfy the *minsup* threshold, therefore, those are discarded. In this fashion Apriori algorithm works to generate candidates by extending one more item and prune in-frequent itemsets. In running example only single itemset of *length-3* items is generated (i.e., `ABC` with support=2), no other itemset with *length-3* is generated because they all do not comply with a given minimum threshold and thus discarded. Algorithm stops when no further candidates are generated.

The bottleneck of Apriori algorithm is generation of candidates; a large number of candidates is generated about *2n-1* when itemset is of size *n*. This is infeasible for memory allocation. Moreover, other limitation of algorithm is its database scanning; that is *n+1* times scanning is needed when the size of longest itemset is *n*. Apriori is infeasible for long sequence patterns and larger databases.

2.4.1.2 Closed association rules

A closed association rule is a comprehensive representation of a set of association rules and it is extracted only from frequent closed itemsets (see Definition 2.4.8) instead from all the itemsets [149]. The closed association rule is described in Definition 2.4.9.

Definition 2.4.8 Frequent closed itemset. *Let $\mathcal{I}$ be a frequent itemset. A frequent itemset $\mathcal{I}'$ is a frequent closed itemset if none of its immediate supersets have same frequency as the $\mathcal{I}'$.*

Table 2.16: Exam correlations (diabetic database)

Exam correlations	*Support (%)*	*Confidence (%)*	*lift*
{Gluocse, Capillary blood, Venous blood} ⇒ {Urine Test}	64	100	1.32
{Capillary blood, Venous blood} ⇒ {Urine Test}	65	99	1.32
{Hb - Glycated hemoglobin} ⇒ {Venous blood}	45	96	1.22
{Urine Test, Venous blood} ⇒ {Glucose}	65	99	1.17
{Glucose} ⇒ {Urine Test}	75	88	1.17
{Venous blood} ⇒ {Urine Test}	65	82	1.09

Definition 2.4.9 Closed association rule. *Let $\mathcal{F}$ be the collection of all frequent items and $\mathcal{F}'$ the set of all frequent closed items where $\mathcal{F} \sqsubseteq \mathcal{F}'$. A closed association rule $\mathcal{R}'$ is a comprehensive representation of set of association rules such that $A \Rightarrow B$, where $A,B \in \mathcal{F}'$.*

The correlation between various exams provides insight information to understand the level of association between the specific exam sets.

2.4.2 Experimental results

The closed association rule mining technique is used for extracting the exam correlations from three real databases and are reported in the following.

2.4.2.1 Case study - Diabetic sequence database

The correlations among diabetic physical examinations mostly appeared between Glucose, Venous blood, Capillary blood and Urine test. These exams are usually diagnosed to monitor the glucose level in blood. Thus, the exam correlations are coherent to medical guidelines. The most frequent correlations are reported in Table 2.16.

Apart from the exams correlations reported in Table 2.16, some other interesting correlations include {Glucose} ⇒ {Venous blood} (support 73% confidence 86% lift 1.08), {Glucose} ⇒ {Capillary blood} (support 75% confidence 88% lift 1.17), {Urine Test} ⇒ {Glucose} (support 75% confidence 100% lift 1.17). The confidence 100% shows the strong correlation between Glucose and Urine Test exams, whilst the correlation {Glucose} ⇒ {Urine Test} has confidence 88% indicating the difference of relationship between both exams. The patients who are diagnosed Urine Test exam first are also been test Glucose level exam, where as patients who are diagnosed Glucose first, 12 out of 100 are not diagnosed Urine Test exam.

Table 2.17: Exam correlations in $Segment_1$ (colon-cancer database)

Exam correlations	*Support (%)*	*Confidence (%)*	*lift*
{Electrocardiogram} ⇒ {Colonoscopy}	56	100	1.14
{Colonoscopy} ⇒ {Electrocardiogram}	56	64	1.14
{Abdomen Ultrasound, Electrocardiogram} ⇒ {Colonoscopy}	25	100	1.14
{Biopsy, Routine Chest Radiograph} ⇒ {Colonoscopy}	28	100	1.14
{Abdomen Ultrasound} ⇒ {Colonoscopy}	37.5	92	1.05
{Closed Biopsy} ⇒ {Colonoscopy}	53	89	1.02
{Colonoscopy} ⇒ {Closed Biopsy}	53	71	0.94
{Abdomen Ultrasound} ⇒ {Closed Biopsy}	25	62	0.82

Table 2.18: Exam correlations in $Segment_2$ (colon-cancer database)

Exam correlations	*Support (%)*	*Confidence (%)*	*lift*
{Closed Biopsy} ⇒ {Colonoscopy}	32	92	2.01
{Colonoscopy} ⇒ {Closed Biopsy}	32	71	2.01
{Closed Biopsy, CAT Scan of Abdomen} ⇒ {Colonoscopy}	32	92	2.01
{Electrocardigram} ⇒ {Colonoscopy}	35	93	1.22
{Colonoscopy} ⇒ {Electrocardigram}	41	88	1.16
{Abdomen Ultrasound} ⇒ {CAT Scan of Abdomen}	37.5	100	1.0
{Electrocardigram} ⇒ {CAT Scan of Abdomen}	75	100	1.0
{Closed Biopsy, Colonoscopy} ⇒ {CAT Scan of Abdomen}	32	100	1.0

2.4.2.2 Case study - Colon-cancer sequence database

The colon-cancer sequence database has been segmented to group patients with similar behaviours. The four segments obtained from colon-cancer database are characterised in Table 2.5. The interesting exam correlations among the exams of $Segment_1$ and $Segment_2$ of colon-cancer sequence database are reported in Table 2.17 and Table 2.18 respectively.

The correlations {Electrocardiogram} ⇒ {Abdominal X-Ray} with (support 74%, confidence 100%, lift 1), and {Abdomen Ultrasound, Abdominal X-Ray} ⇒ {Routine Chest X-Ray} having support 46%, confidence 100% and lift equals to 1.08 are found in $Segment_3$. The $Segment_4$ also represents some exam correlations such as {Abdomen Ultrasound, Electrocardiogram} ⇒ {Routine Chest X-Ray} (support 11%, confidence 100%, lift 2.71) and {Routine Chest X-Ray} ⇒ {Electrocardiogram} with support 34%, confidence 65% and lift 1.45.

The negatively correlated exams indicate weak relationship between them. For example, {CAT Scan of Abdomen, Biopsy} ⇒ {Routine Chest X-Ray} with measures (support 27%, confidence 77% and lift 0.91), the same exams in correlation {Biopsy} ⇒ {CAT Scan of Abdomen, Routine Chest X-Ray } have strength support 27%, confidence 77% and lift 0.91 in $Segment_2$. Simi-

Table 2.19: Exam correlations in $Segment_{Full-Period}$ (pregnancy database)

Exam correlations	*Support (%)*	*Confidence (%)*
{Ultrasonography}⇒{Urinalysis}	81	91
{Glucose}⇒{CBC}	79	99
{CBC, AST}⇒{ALT}	77	100
{AST, ALT}⇒{CBC}	77	100
{Toxoplasma}⇒{CBC}	75	98
{Toxoplasma}⇒{Ultrasonography}	69	90
tabularnewline {Ultrasonography, Glucose}⇒{Urinalysis}	68	97
{HBV}⇒{CBC}	67	100
{Urinalysis, HBV}⇒{CBC}	66	100
{CBC}⇒{CBC, Urinalysis}	66	99
{Toxoplasma, Glucose}⇒{Urinalysis}	65	99
{Glucose, HIV}⇒{CBC}	58	100
{Urinalysis, HCV}⇒{CBC}	54	100
{CBC, Partial thromboplastin time}⇒{Prothrombin time}	49	99
{Prothrombin time, Partial thromboplastin time}⇒{CBC}	49	100
{Prothrombin time, Partial thromboplastin time, Urinalysis}⇒{CBC}	48	100
{Prothrombin time , Partial thromboplastin time, Glucose}⇒{CBC}	47	100
{CBC, Prothrombin time, AST}⇒{ALT}	47	100
{CTG}⇒{Ultrasonography}	42	91
{CBC, Glucose, Rubella Virus Antibody}⇒{Urinalysis}	42	100
{Rubella virus antibody, HIV}⇒{CBC, Urinalysis}	41	100
{Antithrombin III, ALT, AST}⇒{CBC}	41	100
{Antithrombin III, Glucose, Urinalysis}⇒{CBC}	41	100
{ALT, Total Bilirubin}⇒{CBC}	40	100
{Antithrombin III, Ultrasonography}⇒{CBC}	40	100

larly, the exam correlations {Biopsy} ⇒ {Colonoscopy} and {Colonoscopy} ⇒ {Biopsy} having support 62% each, confidence 83% and 71% respectively and lift 0.95 each, and {Colonoscopy, Abdomen Ultrasound} ⇒ {Biopsy} with support 22%, confidence 88% and lift equals to one are observed in Segment_1.

2.4.2.3 Case study - Pregnancy sequence database

The pregnancy sequence database, containing a total of 905 patients has been segmented (see Section 2.3.3.4) into three segments. The $Segment_{Amnio}$ comprises of 73 patients who did Amniocentesis exam - a crucial exam in pregnancy, $Segment_{Non-Amnio}$ contains remaining 832 patients, who did not do Amniocentesis. Another segment $Segment_{Full-Period}$ comprises of 455 patients, all those patients whose complete pregnancy history (i.e., atleast 190 days records) has been available in the considered pregnancy dataset. In addition, patients in $Segment_{Full-Period}$ are further divided into subsets of trimesters. In the following, the exam correlations extracted from each segment are reported.

Table 2.20: Exam correlations in $Segment_{Non-Amnio}$ (pregnancy database)

Exam correlations	*Support (%)*	*Confidence (%)*
{Glucose}⇒{CBC}	62	97
{AST}⇒{ALT}	61	100
{Toxoplasma}⇒{CBC}	59	96
{HBV}⇒{CBC}	51	99
{ALT}⇒{CBC}	51	97
{HBV}⇒{Urinalysis}	50	96
{Urinalysis, HBV}⇒{CBC}	49	99
{Urinalysis, HIV}⇒{CBC}	43	99
{HCV}⇒{HBV}	42	97
{HBV, HCV}⇒{CBC}	41	99
{Urinalysis, HCV}⇒{CBC}	40	99
{CBC, AST, HIV}⇒{ALT}	40	100
{Partial thromboplastin time}⇒{Prothrombin time}	39	99
{CBC, CTG}⇒{Urinalysis}	30	97
{Antithrombin III, Partial thromboplastin time}⇒{Prothrombin time}	30	100

The majority of the exam correlations in $Segment_{Full-Period}$ contains CBC exam as the most frequent exam. Besides, several exam correlations have 100% confidence, indicating that a strong contingency is available between the rule antecedent and consequent. The extracted exam correlations from $Segment_{Full-Period}$ are presented in Table 2.19. The {CBC, AST}⇒{ALT} correlation has confidence of 100% and support of 77%. The support indicates that 77% of patients did all three exams: ALT, AST, and CBC. Whilst, the confidence reflects that all the patients who did CBC and AST, they also did ALT exam. The association between the three exams is reasonable, since these three exams are the part of blood examination. However, some of the association rules (i.e., exam correlations) are not having 100% confidence, even if they should have. For example, {Toxoplasma}⇒{CBC} has 98% of confidence. These unexpected, being exceptional cases refer to specific suspicious conditions in which doctor prescribed Toxoplasma in addition to routinely CBC exam.

The most frequent association rules extracted from $Segment_{Non-Amnio}$, reported in Table 2.20, correspond to rules of $Segment_{Full-Period}$ with the lower support and confidence values. For example, the {Glucose}⇒{CBC} correlation has support 62% and confidence 97% in $Segment_{Non-Amnio}$, whilst the same correlation has support 79% and confidence 97% in $Segment_{Full-Period}$. The diverse behaviour may be due to patients in $Segment_{Non-Amnio}$ may have done a significant number of exams in private structures.

The exam correlations in $Segment_{Amnio}$ are similar to ones found in $Segment_{Non-Amnio}$, but they have the higher values of support and confidence than that of both segments $Segment_{Non-Amnio}$ and $Segment_{Full-Period}$.

However, the availability of Amniocentesis exam in $Segment_{Amnio}$ did not introduce any new exam but increased the strength of the exam correlations. Analogous to $Segment_{Full-Period}$, the correlations in $Segment_{Non-Amnio}$ also have lower frequency indicating exceptional and unexpected cases. For example, the {Partial thromboplastin time}⇒{Prothrombin time} correlation has 99% of confidence value, this means that in 1% of cases, the exam Partial thromboplastin time has been diagnosed without the Prothrombin time. In fact the expected confidence is 100%, since both the exams are done together for measuring intrinsic and extrinsic coagulation pathways.

2.5 Patient clustering

Patients log data are analyzed to identify groups of patients with a similar behaviour in term of performed examinations and highlight the patients who may suffer from additional pathologies. Clustering techniques have been used to discover groups of patients with a similar examination history. The patient groups are then analyzed to extract the medical pathways, characterising these pathways are then assessed with the support of an expert in the medical domain. Identification of the existing categories of a disease is essential to lead towards the success of both treatment procedures and adopted care strategies [157]. The identified subgroups may help to estimate the treatment procedures and resources required for similar groups of patients. Thus, time and costs may be significantly reduced [100]. The following sections describe the clustering techniques briefly and clustering algorithms used in the research.

2.5.1 Clustering techniques

Clustering is the most exploited aspect of data mining. The immense use of clustering is also supported in computer vision, pattern recognition and information retrieval [63]. The clustering process dynamically divides data objects into groups. The goal is to assign objects that are similar (or related) into the same group and objects that are different (or unrelated) into different groups. The more similarity between data objects in a group, greater the homogeneity within the group [153].

A variety of clustering algorithms have been proposed depending upon the nature of data objects, cluster outcomes, available information, objectives of clustering and clustering criteria [63]. A number of techniques of

clustering methods are suggested in [51], [78] and [83]. The clustering techniques can be referred to as supervised and unsupervised. The supervised techniques need some prior information for clustering the data; where as unsupervised techniques need not any prior information about the data. Both terminologies are described in the following.

Supervised clustering The supervised clustering needs some prior information about the data objects in the collection. Based on these information, the data objects are classified into appropriate clusters. This technique acquires prior number of clusters for a given dataset, hence, it works if the information is available. For example, information such as the number of expected clusters, and some attribute label to group together data objects. The supervised clustering algorithm maximizes the purity of data objects, while keeping number of clusters still low [46].

Unsupervised clustering Clustering, usually, is performed when no information is available for the data objects in a given collection. For example, the number of clusters may be unknown. The algorithms that cluster data objects without prior given a number of clusters are sometimes called unsupervised clustering. The aim of unsupervised clustering is to find the correct number of clusters without any prior information about it. Data objects are clustered based on their similarity (or dissimilarity) computed based on a similarity metric (see Section 2.5.3).

The unsupervised clustering is a complex process for issues are to be addressed to obtain some meaningful results. For example, setting the parameters required to carry on clustering process for a given clustering algorithm and evaluating the quality of the clustering results.

2.5.2 Clustering algorithms

The clustering algorithms can be classified into four categories based on their clustering methods: (i) Partitioning, (ii) density-based, (iii) model-based, and (iv) hierarchical-based.

Partitioning methods attempts to decompose a dataset of "n" objects into "k" disjoint partitions, where $k < n$. The general criterion to perform partitioning assigns objects to the same cluster when they are close, and to different clusters when they are far apart with respect to a particular metric. Partitioning methods are able to find only spherical-shaped clusters, unless

the clusters are well separated, and are sensitive to the presence of outliers. K-Means [82] is a popular method which belongs to this category.

Model-based methods hypothesize a mathematical model for each cluster, and then analyze the data set to determine the best fit between the model and the data. These algorithms are able to correctly take into account outliers and noise by making use of standard statistical techniques. Expectation-maximization (EM) [102] algorithm is a representative approach of this class.

Density-based methods are less sensitive to the presence of outliers and identify non-spherical shaped clusters. These methods identify portions of the data space characterized by a high density of objects. Density is defined as the number of objects which are in a particular area of the n-dimensional space. The general strategy is to explore the data space by growing existing clusters as long as the number of objects in their neighborhood is above a given threshold. DBSCAN [50] is a representative algorithm of this class.

Hierarchical-based methods generate a hierarchical collection of clusters by means of either an agglomerative or a divisive approach. The first one, which is the most common, starts with all points as singleton clusters and, at each step, merge the closest pair of clusters. Different cluster proximity measures (e.g., single-link, complete-link, group average) [153] can be exploited to address the merge step. Since the output is a hierachical collection of clusters, these methods are often used when the underlying application requires the creation of a taxonomy/hierarchy.

The problem of analyzing of patients' exam datasets are firstly addressed by exploiting clustering algorithms for identifying different group of patients. The unsupervised algorithms are applied to the medical datasets, considered as case studies in the research, to classify the various groups of patients having similar medical pathways for investigation of a specific pathology. The algorithms used in the research are described in the following sections.

The DBSCAN algorithm

The density based clustering algorithm DBSCAN [50] needs no any prior information for the clustering. It discovers the areas of data objects with the high-density and separates them from the low-density region objects. The DBSCAN clusters each data object based on their center-based density approach. These data objects are classified as core point, border point, and noise point. The core, border and noise points are illustrated in Fig. 2.7. The interior point that falls within a specified distance and its neighbour point

has sufficient neighbours to satisfy the minimum point threshold is called *core point*. A point that has fewer neighbour points within a distance *Eps* but it is still in the neighbourhood of a core point is known as *border point*. Any point, which is neither a *core point* nor a *border point* is referred to as a *noise point* or an outlier [153].

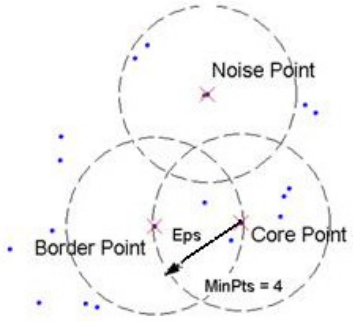


Figure 2.7: Core, border, and noise points [153]

The DBSCAN algorithm requires two parameters called epsilon *Eps* and minimum points *minPts* to place an object into appropriate cluster. The algorithm starts taking unvisited arbitrary point p, then computes the neighbours of this point p where *Eps* value provides the maximum allowed distance (or radius of region) between the points. If the point p has sufficient neighbours i.e., *minPts*, then cluster is formed, else the point is labelled as noise or outlier. The *Eps* or radius is an essential parameter for the DBSCAN, since too large radius may lead to single cluster i.e., all point in a one cluster, or too small radius may cause all point outside the region i.e., outliers. The adequate selection of parameters for the DBSCAN algorithm are described later in section.

DBSCAN can discover arbitrarily shaped clusters and identify outliers as objects in a low density area in the data space. The effectiveness of DBSCAN is affected by the selection of the *Eps* and $MinPts$ values. The time complexity of DBSCAN algorithm is $O(n \times$*time to find points in Eps-neighbourhood*$)$, where n is the number of data points. In the worst case scenario, complexity would be $O(n^2)$. Dense databases may require longer time for DBSCAN computation, it works smoothly for smaller datasets. Moreover, the DBSCAN algorithm specified in Algorithm 3 [153] [39] is relatively sensitive while dealing with databases having arbitrary shapes and sizes, i.e. variation in data objects' densities. It also may cause trouble while handling high-dimensional data objects.

Algorithm 3 `DBSCAN` algorithm [153] [39]

– *DBSCAN($\mathcal{D}$, Eps, minPts)*
Require: *A database $\mathcal{D}$, Eps and minPts*
Ensure: *Clusters (C_k)*
1: $C_k = \phi$
2: **for** *each object obj in D* **do**
3: **if** *obj is not visited* **then**
4: *Mark obj as visited*
5: *Assign neighbours of obj to NP within Eps*
6: **if** $NP \geqslant minPts$ **then**
7: *Create a new cluster C_i*
8: *obj is included in a C_i*
9: **for all** *points p in NP* **do**
10: $Neighbours = neighbours(p, \mathcal{D}, Eps, minPts, C_i)$
11: *Include all Neighbours also in the C_i*
12: **end for**
13: **end if**
14: **else**
15: *Mark obj as noise*
16: **end if**
17: **end for**
18: *Return C_k*
– *neighbours(p, $\mathcal{D}$, Eps, minPts, C_i)*
Require: *Database $\mathcal{D}$, point p Eps minPts and C_i*
Ensure: *Neighbour points of p in $\mathcal{D}$*
1: **for** *each p' in $\mathcal{D}$* **do**
2: **if** *p' not visited* **then**
3: *Mark p' visited*
4: *Assign neighbours of p' to NEP within Eps*
5: **if** $NEP \geqslant minPts$ **then**
6: *Assign NEP to PTS*
7: **end if**
8: **end if**
9: **if** *p' is not memeber of any C_k* **then**
10: *Assign p' to C_i*
11: **end if**
12: **end for**
13: *Return neighbours PTS*

Selection of the DBSCAN parameters The selection of *Eps* and *minPts* parameters for the DBSCAN algorithm is a difficult task. The dataset density is preliminarily analyzed using the k-dist graph [153] to select the *Eps* and $MinPts$ values.

For each object in the collection, the k-dist graph plots the distance to its k^{th} nearest neighbour. On the x-axis objects are sorted by the distance to the k^{th} nearest neighbour, while on the y-axis distances to the k^{th} nearest neighbour are reported. The k value corresponds to the $MinPts$ parameter. The y-axis represents possible values of the *Eps* parameter. By cutting the graph at a given value on the y-axis, the corresponding K value on the x-axis partitions the object collection into the following two subsets as shown in Fig. 2.8. The objects placed on the left hand side of K are labelled by DBSCAN as core points, and those on the rigth side of K as outlier or border points. Sharp changes in the k-dist graph identify dataset portions with a different density [153]. The best selection of parameters helps to get

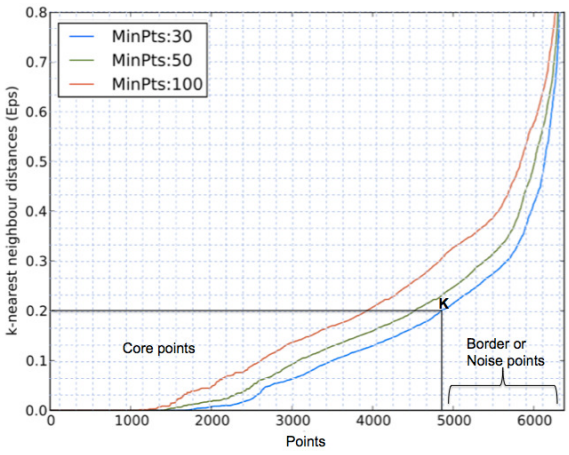


Figure 2.8: k-dist plot of arbitrary data-points

good clustering results. The issue is the determination of *Eps* and *minPts*. The simplest approach that helps to select adequate parameters looks at the distance from a point to its k^{th} nearest neighbour, referred to as k-dist [153].

The Agglomerative hierarchical algorithm

The hierarchical clustering [81] produces a collection of nested clusters arranged into a hierarchical tree called *dendrogram*. Each node in the tree is

a cluster except the root nodes, which contains all data points. The hierarchical clustering can be applied by one of the two approaches, *agglomerative* and *divisive*. The agglomerative (combination) begins with considering each point as an individual cluster and, at each step, merges the closest pair of clusters such as bottom-up approach. The divisive approach starts with one single cluster including all points and, at each step, splits the cluster till all points become individual clusters.

The main steps of the the Agglomerative hierarchical algorithm are reported in *Algorithm 4*. It starts with considering data points as individual clusters then, at every step, the closest pair of clusters are merged into a single cluster [153]. This process needs the definition of *cluster proximity*. The similarity between clusters is referred to as cluster similarity or distance and is computed based on several *cluster proximity* methods, such as single-linkage [141], complete-linkage [87], and average-linkage. The single-linkage or minimum method considers the shortest distance between two clusters (i.e., among their data-points). Likewise, complete-linkage or maximum method considers maximum distance and average-linkage considers the mean distance. The distance or similarity may be computed by any of the distance metric measures [153].

Algorithm 4 Agglomerative hierarchical algorithm [153]

Require: $A\ of\ set\ data-points\ \mathcal{D}$
Ensure: $Clusters\ C_k$
1: $Compute\ proximity\ matrix\ (i.e.,\ singel-,\ complete-,\ average-link)$
2: **repeat**
3: $\quad Merge\ the\ closest\ two\ clusters\ (i.e.,\ points)$
4: $\quad Update\ proximity\ matrix\ reflecting\ new\ proximity\ of\ cluster\ and\ original\ clusters$
5: **until** *only one cluster remains*
6: $Result\ C_k$

The agglomerative hierarchical clustering algorithm requires storage of $\frac{1}{2}m^2$ proximities, where m s the number of data-points. The total number of cluster with m data-points is *m-1*, therefore, in specific cases, the complexity of the algorithm is $O(m^2)$, else, it may be as worsen as $O(m^3)$ [153]. This algorithm is slower and infeasible for large datasets.

The Expectation-maximization algorithm

The Expectation maximization (EM) [102] algorithm finds maximum-likelihood estimate of the parameters of an underlying distribution from a given data set when the data is incomplete or may have missing values.

The EM [102] algorithm is iterative in nature and performs statistical modelling exploiting a gaussian distribution of data. Initially, random values are assigned to parameters. Then, the algorithm repetitively computes parameters till a convergence threshold is reached. Each iteration in the algorithm has two steps: (i) expectation (E) phase, and (ii) maximization (M) phase.
In the E phase, the missing data are estimated given the observed data and the expectation of the likelihood function is updated with parameters computed in previous iterations. In the M phase, the expected likelihood function determined in E phase is maximized to determine new estimates of unknown parameters. These new parameter values are used as input for the new iteration.

For example, consider a set of vectors, where each vector belongs to a gaussian mixture. In other words, set of vectors are picked as samples by one of N gaussian distributions.

2.5.3 Distance measures

The clustering techniques group together similar data objects. Thus, some measurement is required to evaluate the similarity and dissimilarity between the objects. Distance and similarity measures are mainly two measurements to estimate the relation amongst data objects [127]. Based on type and nature of data objects, a variety of distance metrics and similarity measures are suggested.

The distance measures considered in the research to segment the patients' based on the similarity between their medical pathways are defined in the following sections.

2.5.3.1 Euclidean distance

Euclidean distance is distance metric applied on numerical data types. It measures the distance between two points. The greater the distance, the more the farther points are considered. Consider two points x and y, in 1-, 2-,

3- or higher-dimensional space. The Euclidean distance d is mathematically computed [153] as reported in Equation 2.2.

$$d(x, y) = \sqrt{\sum_{k=1}^{n}(x_k - y_k)^2} \tag{2.2}$$

where n is the number of dimensions and x_k and y_k are the k^{th} components of points x and y.

2.5.3.2 Jaccard coefficient

Sneath [140] introduced Jaccard coefficient, which represents maximum cohesion between two objects. The Jaccard coefficient is computed for objects described using binary vector. The binary vector contains only two values 1 and 0, where 1 represents the presence of an attribute for a given object and 0 its absence [128]. The Jaccard coefficient is a similarity measure computed as given in the Equation 2.3.

$$J(a, b) = \frac{\mid a \bigcap b \mid}{\mid a \bigcup b \mid} \tag{2.3}$$

where a and b are two objects of binary values, $|a \bigcap b|$ reports the number of 1's in both objects' binary vector and $|a \bigcup b|$ presents number of 1's in at least one object. The greater the Jaccard coefficient value, greater the similarity between the objects is found. The Jaccard distance is dissimilarity between two objects is measured as:

$$J_\delta(a, b) = 1 - J(a, b) = \frac{\mid a \bigcup b \mid - \mid a \bigcap b \mid}{\mid a \bigcup b \mid} \tag{2.4}$$

A low value on the jaccard distance represents objects with high similarity, and vice-versa. The similarity and dissimilarity values are inversely proportional to each other.

2.5.3.3 Cosine similarity

Cosine similarity is a measure of the angle (i.e., *cosine angle*) between two vectors. It determines the direction of two vectors, whether the two vectors point into same direction (i.e., *angle* 0^0), perpendicular to each other (i.e.,

angle 90^0) or opposite to each other (i.e., *angle* 180^0). This comes into similarity measure function to identify the similarity within data objects (i.e., *vectors*). The cosine measure of vectors x and y is computed by the following Equation 2.5.

$$cosine(x, y) = \frac{x.y}{\sqrt{x.x}\sqrt{y.y}} \tag{2.5}$$

where x and y represents the vectors of patients. The $x.y$ is the dot product (or scalar product) of both vectors. The scalar product is the multiplication of of components of two vectors of equal length. For example, consider $A = \{a_1, a_2, a_3,...,a_n\}$ and $B = \{b_1, b_2, b_3,...,b_n\}$ be vectors of equal length n. The dot product of the vectors A and B, denoted by $A.B$ is computed as:

$$A.B = \sum_{i=1}^{n} a_i b_i = a_1 b_1 + a_2 b_2 + a_3 b_3 + ... + a_n b_n \tag{2.6}$$

The cosine similarity values range in [-1, 0, 1]. The $cosine(x, y)$ equal to 1 describes the exact similarity of vectors x and y. The $cosine(x, y)$ equal to 0 indicates that vectors are independent in nature except the magnitude. The $cosine(x, y)$ equal to -1 points out that opposite directions of the vectors [153].

2.5.4 Clustering evaluation

The evaluation of clustering results to verify the proper assignment of objects to clusters is one of the most difficult tasks. However, several evaluation techniques are suggested in the literature [127]. These criteria generally fall into two categories: *Internal indices* and *External indices*.

Internal indices measure intra-cluster homogeneity, separability or combination of both. These criteria do not need any external information for the computation process. Sum of the squared errors (*SSE*), homogeneity [134], heterogeneity [134] and silhouette [130] are examples of *Internal indices* also called internal quality criterion.

External indices also called external quality criterion examine if clusters match some predefined classification of instances correctly [127]. The rand index [124] is an example of *External indices*.

The evaluation of clusters in the research for the medical data has been carried out using *Internal indices*, since the correct partition (i.e., classifica-

tion or true number of clusters) of the considered data set was not available. In particular, the quality indices: *homogeneity, heterogeneity* and *silhouette* are exploited. These indices are defined in the following sections.

2.5.4.1 Homogeneity

The homogeneity index [134] indicates the compactness between the members of a cluster and is computed as given in the following Equation 2.7

$$Homogeneity_{C_i} = \frac{2}{n(n-1)} \sum_{i=1, x \neq y}^{K} s(x, y) \tag{2.7}$$

where C_i represent a cluster, n and K are total number of members in C_i, and *s(x,y)* is the similarity function between data objects x and y members of cluster C_i. The Equation 2.7 computes the homogeneity value for single cluster C_i, the average homogeneity on the cluster set is the mean value of the homogeneity values of each cluster. The higher homogeneity value represents better clustering.

2.5.4.2 Heterogeneity

The heterogeneity index (or separation index) [134] evaluates the distance between members of the same cluster whether it is lower than the distance between members of different clusters. It is computed as given in the following Equation 2.8.

$$Heterogeneity_{C_{ij}} = \frac{2}{N(N-1) - Q} \sum_{i=1, j=1, i \neq j}^{K} s(x_i, y_j) \tag{2.8}$$

where C_{ij} shows the separation (i.e. distance) of C_i with C_j, N is total number of members in C_i and C_j, K indicates the total number of members in C_i. The Q value represents the number of combination pair on cluster C_i, and is calculated as:

$$Q = \frac{2}{n(n-1)} \tag{2.9}$$

where n is total combination pairs of members in C_i. The similarity function is *s(x,y)*, x_i is the data object belonging to cluster C_i and y_j is the member of cluster C_j.

The Equation 2.8 computes heterogeneity index of a single member of cluster, the average of all members of the same cluster indicates separation

of the cluster. Thus, the mean heterogeneity values of each cluster presents the heterogeneity index on the entire cluster set. The solution improves if the heterogeneity index decreases, thus, lesser the heterogeneity value, the better the clustering results are considered.

2.5.4.3 Silhouette

The silhouette index [130] combines the both similarity and the dissimilarity of an object with its member cluster objects as well as with members of neighbouring clusters. Silhouette values falls in between -1 and 1. The negative silhouette represents wrong placement of the object, where the positive silhouette a better placement of the object [130], and zero silhouette indicates that the object is at the border of cluster. The silhouette index of an object $\boldsymbol{i}$ in a cluster $\boldsymbol{C}$ can be computed as reported in Equation 2.10:

$$Silhouette_i = \frac{b(x) - a(x)}{max\{a(x), b(x)\}} \tag{2.10}$$

where *a(x)* is the average distance of object i with members of cluster C. The *b(x)* is the smallest of average distances of its neighbouring clusters, the distance is computed by means of distance measures such as distance or similarity measures (i.e. eculidean, cosine, etc). The above Equation 2.10 calculates silhouette coefficient for a single data object of cluster C. The average silhouette values for all data objects in the clustering represents the silhouette for that entire cluster C. The mean of each clusters' silhouette indicates the result of entire cluster set.

2.5.5 Experimental results

In this PhD Thesis, the clustering patients (CluP) framework has been proposed to find out groups of patients with similar diagnostic examination history. The patient groups are evaluated with the support of medical experts. In particular, the framework enables (i) characterizing of patients having common physical examinations for a given disease to monitor pathology in standard conditions, (ii) pointing out specific disease complications based on frequent examinations done by patients, and (iii) evaluating the quality of standard care for a given pathology.

The building blocks of the framework are illustrated in Fig. 2.9. They mainly are *Data preprocessing*, *Clustering*, *Validation* and *Domain expert analysis* described in the following sections.

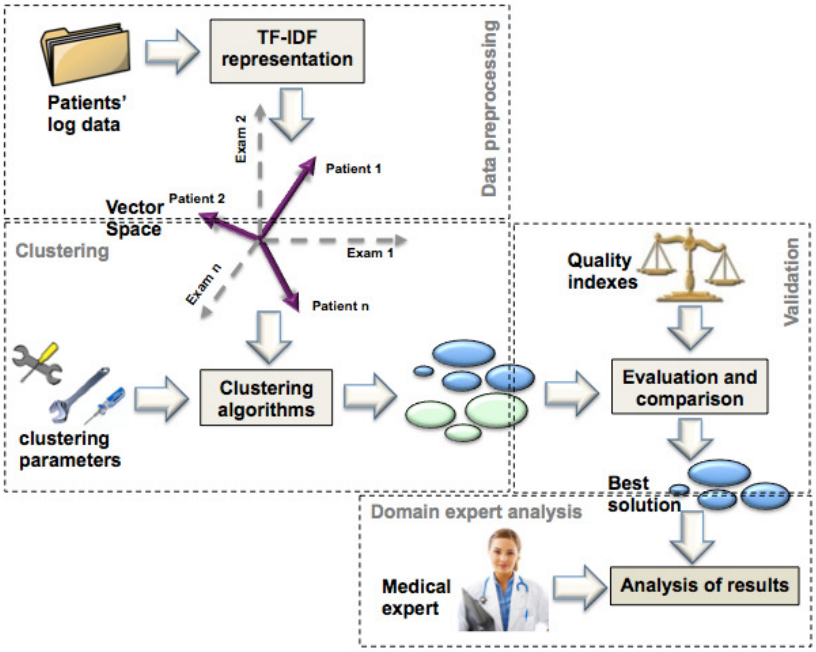


Figure 2.9: The Clustering Patients (CLuP) framework

2.5.5.1 Data preprocessing

This block of the CLuP framework processes the exam log data and represents it into the vector space of examinations done by the patients. Each patient is represented in the examination vector space named *patient vector* (see Definition 2.5.1). Each element in vector is the number of occurrences of the examination done by the patient. Since absolute exam frequency is not suitable to characterize the patients appropriately, weighted exam frequency is represented with the TF-IDF (Term Frequency-Inverse Document Frequency) approach - similar to the approach usually adopted in textual document clustering [145].

Definition 2.5.1 Patient vector. *Let* $\mathcal{P} = \{p_1, p_2, p_3, ..., p_n\}$ *be a collection of patients and* $\mathcal{E} = \{e_1, e_2, e_3, ..., e_n\}$ *the set of examinations done by at least one patient in* $\mathcal{P}$*. A vector* v_i *of* $|\mathcal{E}|$ *cells represents each patient* p_i *in* $\mathcal{P}$*. Each element* $v_i[j]$ *of vector* v_i *indicates the absolute frequency* f_{p_i,e_j} *of examination* e_j *for patient* p_i *as given by*

$$v_{p_i} = [f_{p_i,e_1}, f_{p_i,e_2}, f_{p_i,e_\mathcal{E}}] \tag{2.11}$$

where $f_{p_i,e_j} \epsilon$ *[0, N],* $N \epsilon \mathbb{N}$*, since each patient* p_i *can perform one examination more than once.*

The TF-IDF value increases proportionally to the number of times an examination appears in the patient history, but is offset by the frequency of the examination in the patient collection, which helps to control the fact that some examinations are generally more common than others. Un-weighted examination frequencies do not properly characterize the patient condition, since standard routine tests usually appear with high frequency, while more specific tests may appear with lower frequency.

Representation of patients as weighted frequency vectors The representation of each patient in the dataset requires the computation of the TF and IDF values for each exams done by the patient. The TF-IDF weight of an examination e_j of an arbitrary patient p_i is computed as the product of TF (Term Frequency) and IDF (Inverse Document Frequency) terms as the following:

$$w_{p_i,e_j} = TF_{p_i,e_j} * IDF_{e_j} \tag{2.12}$$

The TF value of pair (p_i, e_j) is the relative frequency of exam e_j for patient p_i, and it is given by:

$$TF_{p_i,e_j} = \frac{f_{p_i,e_j}}{\sum\limits_{1 \leq k \leq |\mathcal{E}|} f_{p_i,e_k}} \tag{2.13}$$

where f_{p_i,e_j} is the number of times a patient p_i has performed exam e_j and $\sum\limits_{1 \leq k \leq |\mathcal{E}|} f_{p_i,e_k}$ is the total number of exams done by the patient p_i. The IDF value of exam e_j shows the frequency of e_j in the patient collection and is computed as:

$$IDF_{e_j} = log(\frac{|\mathcal{P}|}{|p_k \epsilon \mathcal{P} : f_{p_k,e_j} \neq 0|}) \tag{2.14}$$

where $|\mathcal{P}|$ is the number of patients in collection and $|p_k \epsilon \mathcal{P} : f_{p_k,e_j} \neq 0|$ is the number of patients who did at least one exam e_j in collection. The *weighted frequency vector* is the final representation of p_i in vector space model and formally described in Definition 2.5.2.

Definition 2.5.2 Weighted frequency vector. *Let $\mathcal{P}$ be collection of patients and $\mathcal{E}$ the set of exams done by at least one patient in $\mathcal{P}$. Let wv_{p_i,e_j} be the TF-IDF based weighted frequency of exam $e_j \epsilon \mathcal{E}$ for a patient $p_i \epsilon \mathcal{P}$. A weighted frequency vector wv_{p_i} of $|\mathcal{E}|$ cells presents the TF-IDF weights of patient $p_i \epsilon \mathcal{P}$ as:*

$$wv_{p_i,e_j} = [w_{p_i,e_1}, w_{p_i,e_2}, ..., w_{p_i,e_k}] \tag{2.15}$$

where $e_j \epsilon \mathcal{E}$ and $k = |\mathcal{E}|$.

Each patient within the dataset is represented as *weighted frequency vector* for further processing.

The TF-IDF weight wv_{p_i,e_j} for the pair (p_i, e_j) is high when examination e_j appears with high frequency in patient p_i and low frequency in patients in the collection. When examination e_j appears in more patients, the ratio inside the IDF's log function approaches 1, and the IDF_{e_j} value and TF-IDF weight wv_{p_i,e_j} become close to 0. Hence, the approach tends to filter out common examinations.

2.5.5.2 Clustering

The *weighted frequency vectors* are clustered into similar subgroups reflecting specific subtypes of exams using different clustering algorithms (see Section 2.5.2). More specifically, the DBSCAN algorithm [50], the agglomerative hierarchical algorithm [81], and the Expectation maximization (EM) [102] algorithm. Clustering requires metric measurements to compute similarity or distance between objects in the dataset (see Section 2.5.3). The CluP framework considers two patients are similar when they show a similar examination history. The similarity between two patients is measured by the cosine similarity, since the examination history of each patient is modelled as a *weighted frequency vectors.*

Definition 2.5.3 Clustering weighted frequency vectors. *Let* $\mathcal{D} = \{v_1, v_2, v_3, ..., v_n\}$ *be the collection of patients represented in weighted frequency vectors. The clustering algorithm partitions patients (i.e., vectors)* v_i *in* $\mathcal{D}$ *into* K *groups (i.e., clusters)* C_j*, such that* $\bigcup_{1 \leq j \leq k} C_j = \mathbb{V}$ *and* $C_i \bigcap C_j = \phi$*,* $j \neq i$*. Each cluster comprises of* ***similar*** *patients, whilst different clusters contain* ***dissimilar*** *patients.*

2.5.5.3 Validation

The achieved clustering results are evaluated in the *Validation* block of the CluP framework by means of quality indexes (see Section 2.5.4). In particular, the CluP considers *Silhoutte* as a reference quality index. This index balances both the intra-cluster homogeneity and the inter-cluster separation (heterogeneity). The best clustering solution among the evaluated solutions is selected as the final result of the validation phase.

2.5.5.4 Domain expert analysis

The best solution selected in the *validation* phase is investigated with the support of a domain expert to assess the meaning of obtained clusters, identify patients' behaviours in each cluster and derive conclusions about the considered pathology. To provide the characterization of patients in each of the clusters, the most frequent exams in each cluster are extracted.

Details of the results of the application of the CluP framework to a real data are given in the following.

2.5.5.5 An application of the CluP Framework

Experiments are carried out on *diabetic* exam log data (see Table 2.4), which demonstrate the effectiveness of the proposed CluP framework. The three clustering techniques (i.e., DBSCAN, agglomerative hierarchical and EM algorithms) have been applied to get the diversified medical pathways of the considered pathology.

Analysis using DBSCAN algorithm

The most difficult task in using DBSCAN alogrithm is the selection of DBSCAN parameters, thus, k-dist approach (see Section **??**) has been exploited to choose the best parameters. The parameters are tuned to figure out the best configuration according to three values: (i) good values of silhouette (silhouette of at least 0.5) for each obtained cluster, (ii) least possible data fragmentation (i.e., clusters with not very small size, since subgroup models the behaviour of patients' treatment), and (iii) small number of outliers.

The number of clusters obtained and silhouette values at ϵ ranges in [0.2, 0.4], when minPts=30 and DBSCAN algorithm is applied are reported in Table 2.21. The ϵ=0.3 and $minPts$=30 configuration provides the best performance according to the three tuning parameter values in the experimental results. Therefore, the disease complications and diversity of the medical pathways have been detected form the clusters obtained by means of DBSCAN algorithm at these parameters.

The DBSCAN algorithm distinguishes the outliers (dispersed) from the patients of other clusters having similar characteristics and peculiar examinations. Since, a large number of examinations (i.e., 159 exams) are performed by significantly less number of patients, therefore, the patients in outliers are

Table 2.21: Silhouette values for clusters obtained by DBScan algorithm when ϵ varies in the range [0.2, 0.4], minPts=30

	ϵ= 0.2		ϵ= 0.25		ϵ= 0.3		ϵ= 0.35		ϵ= 0.4	
N	$\|P\|$	S	$\|P\|$	S	$\|P\|$	S	$\|P\|$	S	$\|P\|$	S
1	1515	0.87	1522	0.89	1764	0.55	2364	0.31	2707	0.28
2	155	0.65	213	0.62	223	0.67	42	0.95	59	0.91
3	136	0.77	137	0.80	140	0.79	92	0.61	184	0.64
4	185	0.83	212	0.44	294	0.65	173	0.71	52	0.90
5	209	0.58	292	0.64	144	0.74	58	0.59	503	0.38
6	52	0.86	144	0.73	110	0.99	296	0.72	111	0.99
7	76	0.98	109	1.00	42	0.95	148	0.73	41	0.93
8	109	1.00	34	0.97	43	0.85	110	0.99	45	0.94
9	88	0.57	31	0.97	34	0.97	49	0.91	41	1.00
10	43	0.96	41	1.00	36	0.90	40	0.95	30	0.66
11	32	0.98	30	0.88	41	1.00	41	1.00	-	-
12	41	1.00	-	-	-	-	-	-	-	-
Outliers	3739	-0.11	3615	-0.08	3509	-0.33	2967	-0.39	2607	-0.39

N=Number of cluster, $|P|$=Total number of patients, S=Silhouette value

dispersed. To highlight them deeply, the outliers are applied again DBSCAN algorithm by varying parameters till the best solution is obtained. In other words, the experiments have been carried out in multi-levels. The first-level provides the clusters of the analyzed dataset as a whole, whilst second-level presents the clusters obtained from outliers of the first-level, and so on. The process has been terminated until no best solution is detected form the outliers. In the following, frequent examinations extracted from each of the clusters for every level are described.

First-level cluster set

Table 2.22 and Table 2.23 present a comparison among percentages of examinations performed by patients in each cluster, when DBScan clustering algorithm has been applied. The discovered clusters (i.e., approximately 45% of total dataset) can be divided into two groups: Patients having (i) standard tests (i.e., routine) to monitor diabetes conditions (Clusters C_1^1 - C_5^1 in Table 2.22), (ii) coupled with tests to diagnose basic disease complications (Clusters C_6^1 - C_{11}^1 in Table 2.23). The notation C_j^i represents that the *i-th* level of the cluster set and j indicates the local cluster number. The cluster characteristics are described in details in the following.

In the two largest clusters C_1^1 and C_2^1, patients performed standard rou-

Table 2.22: Exam frequencies (%) in first-level cluster set with routinely tests

Category	*Examination*	C_1^1	C_2^1	C_3^1	C_4^1	C_5^1
Routine	First visit	-	13	100	-	-
	Visit	78	96	58	100	100
	Glucose level	78	98	63	-	100
	Urine test	72	97	58	-	-
	Venous blood	96	75	35	-	-
	Capillary blood	72	97	58	-	-
	Haemoglobin	100	-	-	-	-
Cardiovascular	Electrocardiogram	-	-	100	-	-
Eye	Fundus Oculi	-	-	28	-	-
Number of Patients		223	1764	43	110	41
Silhouette		0.67	0.55	0.85	0.99	1.0

tine tests of diabetes. The patients in C_3^1 besides routinely tests have been tested with exams (usually done at the beginning) to diagnose the most frequent diabetes complications (mainly risks for cardiovascular disease and eye problems). Differently from cluster C_3^1, patients in C_2^1 have been tested few "First visit" without any additional tests such as Electrocardiogram.

The specialistic visits are carried out by patients in clusters C_4^1 and C_5^1, whilst, glucose level test is also done in C_5^1. These clusters may include patients usually tested in private structures, and periodically reporting test results to the sanitary agency.

Patients in clusters C_6^1 - C_{11}^1, have been tested with additional tests to diagnose diabetes complications on (i) eye (C_6^1), (ii) cardiovascular system (C_7^1), (iii) both eye and cardiovascular system (C_8^1), (iv) carotid (C_9^1), and (v) limb (C_{10}^1). Finally, (vi) cluster C_{11}^1 includes tests for liver, renal, and in particular cardiovascular. Differently from the cluster sets C_1^1 - C_5^1, diabetes complications have been monitored in clusters C_6^1 - C_{11}^1 through few (quite standard) tests, showing a limited degree of seriousness. Only in cluster C_{10}^1 the cardiovascular system has been deeply tested. Standard routinely examinations still characterize clusters C_6^1 - C_{11}^1 even though they appear with a lower frequency than in clusters C_1^1 - C_3^1.

Patients with exam history significantly dissimilar to all the others are labelled as outliers and are not included in any cluster in Table 2.22 and Table 2.23. The DBSCAN algorithm has been re-applied on only these 3509 patients (approximately 55% of total dataset), with different parameter values to deeply investigate their disease complications. Results are discussed in the following section.

Table 2.23: Exam frequencies (%) in first-level cluster set with complications

Category	*Examination*	C_6^1	C_7^1	C_8^1	C_9^1	C_{10}^1	C_{11}^1
Routine	First visit	-	-	-	-	-	-
	Visit	77	78	66	62	68	97
	Glucose level	74	74	64	62	59	97
	Urine test	74	74	64	57	56	92
	Venous blood	57	60	53	48	44	97
	Capillary blood	74	73	63	55	56	92
	Haemoglobin	-	-	-	14	12	100
	Complete blood count	-	-	-	-	3	3
Cardiovascular	Electrocardiogram	-	100	100	-	-	42
	Cholesterol	-	-	-	-	-	100
	HDL Cholesterol	-	-	-	-	-	100
	Triglycerides	-	-	-	-	-	100
Eye	Fundus Oculi	100	-	100	-	-	39
Liver	ALT	-	-	-	-	-	100
	AST	-	-	-	-	-	100
Renal	Culture urine	-	-	-	-	-	100
	Creatinine	-	-	-	-	-	3
Carotid	ECO doppler carotid	-	-	-	100	-	-
Limb	ECO doppler limb	-	-	-	-	100	-
Number of Patients		294	144	140	42	34	36
Silhouette		0.65	0.74	0.79	0.95	0.97	0.90

Second-level cluster set

Patients analyzed at this level are characterized by more diversified exam histories with DBSCAN parameters ϵ=0.4 and $minPts$=30. More specifically, the following two main categories can be identified. (i) Patients tested with peculiar tests to diagnose a specific diabetes complication (clusters C_1^2 - C_2^2). (ii) Patients who undergone various tests to diagnose different diabetes complications (clusters C_3^2 - C_5^2). These categories respectively indicate patients seriously affected by a particular disease problem or suffering from more disease problems at the same time. The second-level cluster set is reported in Table 2.24 and discussed below.

Clusters (C_1^2 and C_2^2) include peculiar exams to diagnose eye problems. More specifically, all patients in cluster C_1^2 did the tests to assess vision and ability to focus on objects (called "Eye examination" in Table 2.24). Instead, all patients in cluster C_2^2 had Retinal photocoagulation, a laser operation done in cases of long-term eye complications, such as the proliferative retinopathy.

All patients in clusters C_3^2 - C_5^2 suffer complications on cardiovascular, liver, and renal system, but with different degree of seriousness. Patients in cluster C_5^2 show liver complications with higher gravity than that of clusters C_3^2 and C_4^2, since their exam history includes more tests in this category.

Table 2.24: Exam frequencies (%) in second-level cluster set with more serious diabetes complications

Category	*Examination*	C_1^2	C_2^2	C_3^2	C_4^2	C_5^2
Routine	First visit	-	13	-	-	-
	Visit	65	90	22	98	71
	Glucose level	52	92	22	100	95
	Urine test	37	90	19	98	68
	Venous blood	35	84	100	100	98
	Capillary blood	37	85	17	98	63
	Haemoglobin	13	34	100	98	83
	Complete blood count	3	15	45	7	93
Cardiovascular	Electrocardiogram	17	21	70	47	20
	Cholesterol	7	25	100	97	93
	HDL Cholesterol	4	26	100	99	92
	Triglycerides	7	26	100	97	90
Eye	Fundus Oculi	50	38	53	49	27
	Eye examination	100	5	-	-	-
	Angioscopy	-	30	-	-	-
	Renital photocoagulation	-	100	-	-	-
Liver	ALT	-	21	10	98	98
	AST	-	21	8	97	98
	Bilirubin	-	2	-	-	100
	Gamma GT	-	-	100	-	95
Renal	Culture urine	2	11	67	97	44
	Creatinine	4	20	99	11	78
	Microalbuminuria	-	7	100	60	37
Carotid	ECO doppler carotid	-	5	-	-	2
Limb	ECO doppler limb	-	-	-	-	-
Number of Patients		42	144	294	140	34
Silhouette		0.95	0.74	0.65	0.79	0.97

Relatively, clusters C_3^2 is characterized by more serious renal complications than those of clusters C_4^2, and C_5^2. Cardiovascular complications have similar gravity in C_3^2 - C_5^2 clusters, because exams in this category appear with similar frequency.

At this stage detected 5 homogeneous clusters contain 570 patients (i.e., 9% of total dataset), the remaining 2939 patients (i.e., 46% of total dataset) are classified as outliers and not assigned to any cluster. Since number of patients is quite large, therefore, DBScan algorithm is applied again on them by modifying the parameter settings. Results are analyzed in the following section.

Third-level cluster set

The trend already observed in second-level appear also in this level, because the treatment are formed by more specific examinations (e.g., high level of Angioscopy for eye complications) and patients suffer from more than one complication at the same time (e.g., carotid, liver and cardiovascular complications). At this level, clusters are still homogeneous (approximately 27% of total dataset) according to silhouette index and only 1239 patients (approximately 19% of total dataset) are classified as outliers. The results collected at this stage (with DBSCAN parameters ϵ=0.6 and $minPts$=30) follows the trend analyzed in second-level clusters, (i.e., similar medical pathways). Furthermore, lesser number of outliers are analyzed with highly sparse patterns because 1239 patients are tested 159 distinct physical examinations. The next level of DBScan algorithm adoption did not produce homogeneous clusters due to the fact that large number of distinct exams (i.e., 159) is tested for 1239 patients (i.e., 19% of total dataset), which indicates the non-uniformity among them. Therefore, multi-level clustering process is terminated.

Analysis using Agglomerative hierarchical and EM algorithms

This section reports the clustering results achieved when agglomerative hierarchical and EM algorithms (see Section 2.5.2) are applied on the considered diabetic dataset. To obtain the desired number of clusters in agglomerative hierarchical clustering, flatten clustering operator [105] is exploited, which expands nodes in hierarchical clustering in order of node's distances till the desired number of clusters are achieved. Furthermore, complete-link proximity method (see Section 2.5.2) is applied in hierarchical algorithm, since other proximity methods (i.e., single-link and average-link) did not produce adequate clustering results.

The obtained clusters by varying the desired numbers of clusters through agglomerative hierarchical and EM algorithms, unfortunately, could not produce the best solutions in the validation phase. Table 2.25 reports the clusters achieved using both clustering algorithms (i.e., agglomerative hierarchical and EM) and their corresponding quality indexes in terms of silhouette values as representatives of the other experimental results.

The negative silhouette values of clusters indicate the wrong placement of the patients in them. Thus, the distribution of patients in each cluster and quality indexes disclose the fact that the grouping of patients is inadequate to highlight the complications of disease and dispersed nature of the treatment

Table 2.25: Silhouette values for clusters obtained by Agglomerative hierarchical and EM algorithms

N	Hierarchical algorithm						EM algorithm					
	$\|P\|$	S	$\|P\|$	S	$\|P\|$	S	$\|P\|$	S	$\|P\|$	S	$\|P\|$	S
1	2544	-0.28	2560	-0.27	2583	-0.27	1050	-0.11	464	-0.14	90	0.18
2	30	0.19	89	-0.09	89	-0.09	1798	0.20	2155	0.26	2418	0.47
3	89	-0.09	382	0.13	382	0.13	294	0.12	187	0.15	462	-0.08
4	1624	0.84	1624	0.84	1624	0.84	97	0.02	93	-0.08	387	-0.06
5	118	0.54	30	0.19	30	0.19	219	-0.24	44	0.06	134	0.60
6	382	0.13	685	0.35	685	0.35	140	-0.01	143	-0.07	240	-0.05
7	256	0.47	256	0.47	256	0.47	50	-0.08	63	-0.05	89	-0.06
8	343	0.61	343	0.61	343	0.61	119	0.84	56	-0.03	94	-0.04
9	270	0.10	270	0.10	270	0.10	85	-0.08	81	-0.06	94	-0.06
10	23	0.71	23	0.70	118	0.54	35	-0.10	20	0.15	2372	0.18
11	16	0.47	118	0.54	-	-	72	-0.04	3074	0.33	-	-
12	685	0.35	-	-	-	-	2321	0.51	-	-	-	-

N=Number of cluster, $|P|$=Total number of patients, S=Silhouette value

procedures. Unlike the DBSCAN algorithm, the algorithms hierarchical and EM are more sensitive to outliers. Moreover, the analyzed dataset include sparse data (i.e., outliers) as specific examination pathways for diabetes conditions. Therefore, these algorithms could not produce promising cluster sets. The analysis of the medical pathways to get the diversified knowledge also could not achieved.

Chapter 3

Textual Data Mining

This chapter addresses the textual data analysis issues. Section 3.1 briefly introduces text mining, while Section 3.2 describes the related works. The adopted approaches to address the issues about *text summarization* and *user-generated content* are presented in Sections 3.3 and 3.4 respectively.

3.1 Text mining

Text mining (also called text data mining) addresses the discovery of novel, hidden and unknown information by automatic means from a set of unstructured textual resources, usually textual documents. The automatic information discovery takes place exploiting sophisticated algorithms and techniques.

Text mining is a branch of data mining, which focuses on extracting information from textual documents containing natural language text where as data mining extracts from databases [38]. Though web mining also retrieves information from web documents that may also contain natural language text, but there is significant difference between the two techniques (i.e., web mining and text mining). *Web mining* extracts knowledge from structured web documents unlike text mining, in which textual data is unstructured. Moreover, text mining differs from Information Retrieval (IR) or Information Access. Text mining deviates from information retrieval for IR retrieves no any genuinely novel information like text mining does.

The text mining process comprises of set of sub-processes such as *text preprocessing, text transformation, attribute selection, pattern discovery/data mining* and finally *evaluation or interpretation* as shown in Fig. 3.1. A large

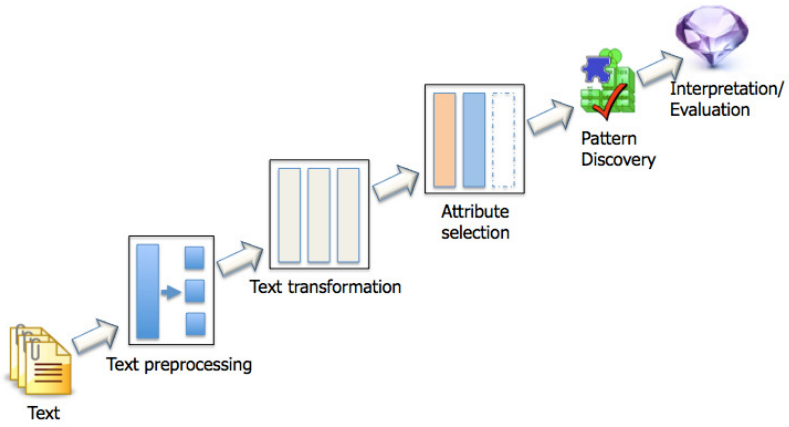


Figure 3.1: Text mining processes [165]

collection of textual data is preprocessed for cleaning text, tokenization and tagging. Once text data is preprocessed, it is represented into word format usually vector space representation or bags of words, this step is referred to as text transformation. Then, features of various textual data are selected at attribute selection step. The irrelevant attributes are ignored and only interested features are selected. Pattern discovery is traditional data mining approach, since the textual data is available in structured format. In the end, interpretation steps evaluates the extracted information for a given problem.

The PhD research addresses the problems of information extraction for human consumption, for instance, text summarization, particularly, multi-document summarization. In addition, user-generated content (UGC) has also been analyzed in the PhD research by exploiting established data mining techniques. In the following sections, a detailed description of the most recent approaches related to knowledge extraction from textual data and user-generated content analysis are presented.

3.2 Related works

Emerging Social Networking allows accessing large amount of textual data. To understand lengthy web documents, automatic text summarization is becoming more and more important issue [55, 64, 80, 96, 107, 159]. Furthermore, established data mining techniques (e.g., traditional and generalized association rule mining) have also been exploited to perform user-generated content analysis. The recent research works carried out in text summariza-

tion and user-generated content analysis are presented in the following.

3.2.1 Text summarization

A number of approaches address the selection of appropriate sentences to summarize textual documents [48, 151, 161], where sentence evaluation is exploited using cluster-based (e.g., [121], [161]) and graph-based (e.g., [48]) approaches. The clustering approaches classify sentences using clustering algorithms and select sentences to include into summary from each cluster. Graph-based approaches, exploiting graph-based models, represent correlations among sentences. Several attempts are made for graph-based ranking methods. For example, [131] used degree centrality to extract important paragraphs in a single document summarization. The work in [106] selects important sentences based on their rankings; they built a graph, where nodes of the graph are sentences in the text and the edges of the graph are sentence inter-connections (function of content overlap). The work in [107] emphasized the need of automatic text summarization. For example, the text summarization work in [13] is carried out using lexical chains, which integrates word-sense to segment related text and extracts sentences. A frequent term based text summarization algorithm is proposed in [108], which is implemented in java programming language. This algorithm is based on three phases. Initially documents are pre-processed to remove unnecessary words and characters, the second phase collects frequent terms and also generates their semantic equivalent terms and in the final phase, the sentences containing frequent and semantic terms are filtered for summarization [108].

The text summarization based on 10-word summaries that produces the most important noun phrases is presented in [16]. [96] presented an approach to select the most important information contained text from the text for the text summarization task. The work also discussed evaluation of the text summarization techniques, suggesting preliminary approach using quality indicators (i.e. summary coherence, identification, informativeness etc) instead of quantitative manner. The work in [159] proposed SUMMTERM system for automatic summarization, the system after preprocessing text, uses term extractor to get ordered list of term candidates contained in the text, finally summary is generated based on terms similarity, and some parameters such as threshold, number of sentences in the final summary, etc. [123] discussed the use of Wikipedia (The World's largest online encyclopedia) for understanding the semantic meaning of words in the text summarization. The WikiSummarizer system reported in TAC 2010 is proposed by

[104], which summarizes the text based on features gathered from Wikipedia. The WikiSummarizer enhances the comprehension of sentences with concepts obtained from Wikipedia called sentence Wikification unlike traditional approaches, which maps TFIDF values of words. The Sentence Wikification represents sentence as a set of Wikipedia concepts. The semantic relatedness of concepts is also taken into consideration in Wikisummarizer System. The work in [80] described summarization technique independent of sentence extraction. The proposed work in [80] generates set of extracts formed by conjunction of sentences from the original text, which are also compared with each other and classified to yield best extracts thus optimal extract is selected.

Besides text summarization, the work of automatic categorization of news is described in [8]. Furthermore, a number of attempts are proposed about evaluation of text summarization [98, 112]. The real-time summaries of events from tweeter tweets are addressed in [27], the approach learns hidden state of events by means of Hidden Markov Models and the provide help in summarizing tweets. The summarization framework named Opinosis is proposed in [58] that produces abstractive summaries from highly similar opinions using graphs that represent natural language text. The framework has not any domain knowledge, though it uses a little knowledge from Natural Language Processing (NLP). User adopted or personalized summary represents the interests of corresponding user. In the user-adopted summaries, the reader's (user's) interests are taken into consideration and are given more importance while summarizing the text instead of frequent items of the text. For instance in [40], the personalized summarization approach is addressed, aiming to provide a user-oriented summary that is interesting to user. A topic retrospection system, which identifies several events from news topics and produces summaries that sketch evolutionary events in the news, is proposed in [93]. The approach comprises of three main functions: event identification, main storyline construction and finally storyline-based summarization. [135] described the hottest issue of today's era namely automatic summarization of tweeter topics. An algorithm is presented which takes general phrase for any user given phrase, based on that phrase; a large number of tweets are collected and an automated summary of the tweets is produced.

The problem of social context summarization for web documents is addressed in [170], The proposed approach combines both web documents and social context into a unified framework and the problem is formulated by means of dual-wing factor graph (DWFG) model. Moreover, the aim of the work in [170] is to build a high quality summary of web documents consider-

ing the social context influence and the proposed method works in supervised manner.

The proposed approach in [115] examines relevant sentences and identifies their importance based on higher scores by means of clues: *Frequency-keyword approach, title-keyword method, location method, syntactic criteria, clue method, relational criteria* and *Indicator phrase method*. [174] identifies events and rank events for a single document; the major analysis focus on construction of event relation graph that describes events and their relation of a document; further event weights are computed by menas of PageRank Algorithm. [70] introduced a method of multi-document summarization based on graph model. The nodes in the graph represent sentences of documents and edges are similarities between them, finally, PageRank algorithm produced scores for the sentence selection. A graph-based model (i.e. LexRank) is proposed in [48] in which sentence selection for the summarization is based on the concept of eigenvector centrality computed by means of PageRank algorithm [21]. Moreover, the relevant sentences are computed with tfidf scores of the words; the sentences having more number of higher scoring words are selected. [61] presented two methods for text summarization by means of ranking and extracting sentences from original documents; The first method works on sentence relevance while the other works on identification of semantically important sentences using latent semantic analysis.

3.2.2 Analysis and visualization of user-generated content

The birth of social networking has greatly taken attention of researchers to address the significance and effective use of social web data. The recent research investigates the use of social data in e-commerce and discovers the user habits and interests of different geographical online communities to support analysts in decision-making and optimal resource management in business as well as web maintenance. Modern technologies and tools uncover the significance of user related data (i.e. digital footprints, tweets, etc) that leads towards understanding both individual and social behaviours [59].

Social media has provided large amount of user-generated contents; such contents represent vast variety about real-world events, some contents contain information, opinion, event time stamps etc. The user-generated content (UGC) is amongst the most popular contents generated on social networking platform. The search engines and social sciences could increasingly get fruits from such user-generated contents [14] reported approach about find-

ing highly relevant and informative messages from vast variety of Tweeter messages. The approach proposed in [14] aimed at exploring the most useful and informative messages about events. The sentimental analysis of users from tweeter data is addressed in [19] resulting in significant impact by the events in the social, economical, political, and cultural on users' mood.

The usage and understanding of the micro-blogging has been focused in recent years. For example, the work performed in [71] highlighted the twitter as conversational tool for the user to interact and collaborate. [156] reported the use of micro-blogging as the means to predict the political sentiment. The discovered knowledge from the tweets reflected the people sentiments for the correspondent political parties of the German federal elections. Some researchers are active in the analysis of product marketing based on twitter posts, for instance, the work in [79] claimed the customer sentiments derived from their tweets regarding branding of the products as the opinion of the customers for the brand. The proposed visualization tool in the PhD research activity targets the analysis of Twitter textual content.

Two established data mining techniques are particularly suitable for analyzing user-generated contents: (i) data visualization and (ii) generalized association rule mining.

3.2.2.1 Visualization

A significant research effort has been devoted to visualize the several well-known KDD tasks, which help to figure out valuable information as well as support the analysts in decision-making processes. For example, classification of information visualization is presented in [85]. A number of research work has also addressed smart data visualization (e.g., [68],[85],[86]), whilst some research work (e.g., [88],[90],[168]) has been carried out focusing on visualizing patterns that are discovered using general purpose data mining approaches. The analysis of animal movements is demonstrated in [90]. In addition, the pattern discovered by means of pattern mining and trajectory analysis are embedded on visualization tools such as Google Maps, or plotted on 2-Dimensional plane enriched with available statistics to highlight the deep insights of the discovered knowledge. Modelling of correlations amongst data entities in association rule mining facilitate the users to understand relationship of various interdependencies of data deeply. The tools to visualize association rules are proposed in [28, 162]. Although several modelling and visualization tools are developed, yet majority of the tools represent not more than few dozens of rules [168]. The commercial visualizing tools, for example,

can be found in MineSet [133] and QUEST [75]. The visualization techniques are categorized as grid view, information landscape view and node link view to represent the association rules [95].

The screen is framed into smaller cells corresponding rules in grid view. Item-to-item grid view and support-to-confidence grid view are proposed in [74]. The information landscape grid is 2D grid approach, with two methods: item-to-item and rule-to-item representations [168]. Node link view, self-explanatory name, represents association rules as nodes and links. Nodes represent items and links relation between the nodes. The color and width of the link indicates corresponding measurements of the rule [95]. MIRAGE (Interactive Graphical Exploration of Minimal Association Rules) represents interactive graphical visualization of minimal rules in comprehensive format. The MIRAGE framework is capable of taking all frequent or closed frequent itemsets along with their support threshold and creates lattice having closed set as a node [172]. Domingues et al. [42] introduces GART algorithm (Generalization of Association Rules using Taxonomies) and RulEE-GAR computational method to generalize and analyze the generalized rules. The GART algorithm takes one side of the rule (i.e LHS or RHS), the rules are grouped into subsets, which present equal antecedent or consequents. In case LHS is taken for generalization, the subsets would be generated by means of RHS and vice-verssa; moreover, subsets are generalized using taxonomies and rules are stored as generalized association rules; where as RulEE-GAR method analyzes the generalized rules [42].

C_BVAR (Clustering-based Visualizer of Association Rules) prototype to visualize rules is presented in [37]. The prototype used meta-knowledge, which helped in exploring rules, during clustering to assist association rules. [103] introduced a tool called Web Visualization System (WEBVS) containing visualizing and fuzzy association rule mining as an integral part of e-Government portal. The WEBVS provides visualization of association rules obtained form web logs containing web access data [103]. [18] developed information landscape 3D representation of interesting rules described by various quality measures. The effectiveness of the previously mentioned approaches strongly depends on the analyzed data distribution, since the effectiveness of the approaches is biased by the highly detailed granularity level when textual data is sparse. Moreover, the information at the lowest granularity levels is not easily accessible. The proposed tool in this PhD research specifically addresses generalized rule visualization from textual data (i.e., tweets). The discovered rules are visualized in a graph-based model and their granularity level is selected by the analyst by exploring the input taxonomy.

3.2.2.2 Generalized association rule mining

Association rule mining allows identification and discovery of information based on interested correlations among data. The major drawback of traditional rule mining technique is the strict dependency in business decisions on abstraction level of the analyzed data for the adopted approaches. Sometimes, traditional rule mining algorithms are not effective in mining valuable knowledge, because of the excessive detailed level on the mined information. Particularly, potentially relevant but rare knowledge may be discarded due to the enforcement of the minimum support threshold.

To cope with the traditional rule mining issue, [143] introduced the concept of generalized association rule mining. The generalized association rule is an association rule (see Definition 2.4.1), which presents high level correlations among data. A taxonomy (i.e., a set of is-a hierarchies) aggregates data items into their upper and higher level generalizations; exploiting taxonomy, the generalized association rules are generated by combining items belonging to different abstraction levels. The detected generalized rules may support experts in decision process more robustly and better than that of traditional rule mining approach, since high level view of analyzed data also represents the knowledge covered by their low level infrequent descendants.

The first generalized association rule mining algorithm - Cumulate is introduced in [143] presented, which is an Apriori-based algorithm [2] and for each item, it generates generalized itemsets by considering all its parents in the hierarchy. A more efficient process of extracting generalized association rule mining is based on new optimization strategies [65, 69]. In [69], exploiting TID intersection computation that commonly for rule mining algorithms designed for the vertical data format, a faster support counting mechanism is provided. On the contrary, an optimization technique based on traversal of top-down hierarchy and multiple support threshold is proposed in [65], which identifies in advance generalized itemsets that can not be frequent by means of apriori-based approaches. Recently, [9] proposed an algorithm, which performs support-driven itemset generalization, in other words, a frequent generalized itemset is detected only if it has at least an infrequent (rare) descendant. To restrict the generation of generalized rules at some extend, a confidence-based constraint has also been proposed in [11], which aims at preventing the generation of misleading high-confidence rules and hence, improve the quality and compactness of mining results.

The proposed visualization approach in the PhD research is taxonomy-driven, i.e., the taxonomy used to drive the generalized rule mining process

and focuses on visualization of frequent generalized association rules derived from tweets. The PhD research activity addressing *text summarization* and *user-generated content analysis* is reported separately in the following sections.

3.3 Text summarization

Text summarization aims at saving time by reducing length of textual data and keeping the salient points of the text into resultant summaries of web documents. The automatic summaries may be achieved from a single document or multiple documents of similar topics. For example, extracting essential sentences from textual data using important *words* and *phrases frequency* are addressed in [97]. The summarization approach is termed as keyword-based technique, if it includes salient keywords in generating summary. On the contrary, the approaches that detect the most essential and informative sentences by classifying documents into sentences, and then include these sentences into summary are referred to as sentence-based techniques.

The challenge in text summarization lies in the identifying and distinguishing the more essential and informative parts of the textual data from those of less informative ones. Mostly, the work in single-document summarization, emphasizing on *technical documents*, builds a summary from single document [38]. Whilst, multi-document summarization constructs a summary representing most relevant and informative parts of sentences belonging to set of documents. In sentence-based summarization context, several different approaches have been already proposed, which evaluate sentences with respect to clustering-based, probabilistic-based or graph-based (e.g., [60], [121],[151],[155]). The graph-based approaches build a graph in which the nodes are either sentences or terms (i.e., words), and edges represent the strength of association between the nodes. The approach proposed in [151] presented associations among sentences by graph-based model. Moreover, the selection of sentences have been carried out by exploiting eigenvector centrality computed with the help of well-known PageRank algorithm [21]. However, the focus of previous work has been on significance of a single word, whilst correlations among multiple words have not been effectively captured. The novel text summarizer developed in the PhD research activity is presented in the subsequent sections.

3.3.1 Graph-based Summarizer (GraphSum)

A novel graph-based multi-document summarizer - Graph-based Summarizer (GraphSum) developed in the PhD research builds a correlation graph, whose nodes are the set of document terms with arbitrary size and the edges of the graph represent the correlations between the set of document terms weighed by correlation strengths (i.e., support and lift [153]). The frequent terms strongly correlated with each other (either positive or negative) in the analyzed collection are represented by the adopted model. The well-known data mining technique i.e., *association rule mining* [2] has been exploited to discover the correlations among the document terms.

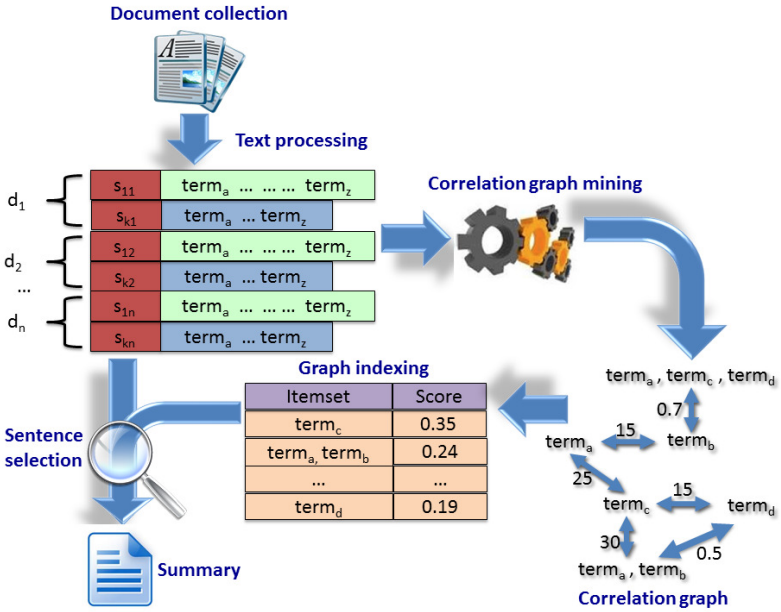


Figure 3.2: The GraphSum summarizer

To ensure quality of the GraphSum summarizer, two constraints are exploited in the mining process (i) A minimum support threshold (*minsup* (i.e., frequency of the considered term in the source data) (ii) A maximum negative and a minimum positive correlation threshold [153] (i.e., significance level of selected positive and negative correlations among set of terms). The GraphSum evaluates graph nodes and selects the most relevant sentences to include in the summary driven by the correlation graph. To achieve this goal, a popular PageRank algorithm [21] is applied in the graph ranking strategy. In addition, to determine the positively and negatively correlated terms in the graph ranking, PageRank score propagation through negatively correlated

links is mitigated. Whilst, the strongly correlated (i.e., positive correlations) terms are more likely to get high ranking, thus, are considered for summary. Instead the negatively correlated terms, on average, are penalized with some of their neighbour terms in the graph. The sentences from the top ranked, are selected and included in the summary, which cover the generated model the best.

The main building blocks of GraphSum are shown in Fig. 3.2. The GraphSum adopts basic preprocessing steps as semantics-based analysis such as stemming and removal of stop-words unlike many other state-of-the-art approaches [36] [35] [146]. Therefore, this yields high probability of set of documents belonging to different contextual application domains. In the following sections, a detailed description of each step of the GraphSum summarizer is given.

3.3.1.1 Text processing

The raw textual data, unsuitable for the mining process, is prepared for the further mining steps. The very basic semantics-based analysis is performed to keep the model's flexibility and usability at higher level in various application domains. Particularly, noisy information have been removed such as numbers, stop words, and URLs. Besides, avoiding noisy information, stemming algorithm based on Wordnet [169] and Snowball [17] is applied to represent the documents into their base roots (i.e., stems). The outcome of the text processing yields the bag-of-word (BOW) representation [152] for the analyzed document collection. Each document $d_k \in D$ comprises of a set of sentences $S_k=\{s_{1k}, \ldots, s_{zk}\}$, where each sentence includes a set of unordered word stems, also called *terms*.

Definition 3.3.1 Transactional representation of a document collection. *Let $D=\{d_1, \ldots, d_N\}$ be a document collection. The transactional representation T of D contains, for every sentence s_{jk} belonging to any document $d_k \in D$, a transaction tr_{jk} composed of distinct terms w_q where w_q is the q-th term in s_{jk}.*

The document collection, after preliminary processing, is modelled as transactional dataset in which each document sentence s_{jk} corresponds to a transaction tr_{jk}, which contains set of distinct terms available in BOW representation of the s_{jk}. Formally it is described in Definition 3.3.1.

For example, consider a document collection D as shown in Table 3.1, which contains three documents. Each document comprises of two sentences, their corresponding transactional representation is reported in Table 3.2. The transactional representations contains 6 transactions, each of them is associated with a distinct document sentence. The corresponding terms for each of the sentences in document collection D in Table 3.1 are presented in Table 3.2, e.g., the first sentence of document d_1 includes the terms *Data* and *Analysis*.

Table 3.1: Document collection D before text processing

Document	Content
d_1	This is about data analysis. In particular, it analyzes contextual information.
d_2	Information is mostly hidden in data. However, through the analysis of the context we may enrich data.
d_3	Processing data is useful: an in-depth analysis produces actionable information.

Table 3.2: Document collection D after text processing

Document	Sentence ID	Sentence
d_1	1	Data, Analysis
	2	Analysis, Context, information
d_2	3	Information, Data, Hide
	4	Analysis, Context, Data, Enrich
d_3	5	Data, Processing, Useful
	6	Depth, Analysis, Information, Action

3.3.1.2 Correlation graph mining

The correlation graph mining block of the GraphSum summarizer makes extraction of most significant hidden correlations in the analyzed document collection. The transactional representation T of considered document collection $D=\{d_1, \ldots, d_N\}$ is an input to this block, which produces a graph G, which is further used for extracting most relevant sentences that will be eventually included in the summary.

Correlation graph mining involves three steps: (i) Frequent itemset mining, (ii) term set correlation estimate, and (iii) graph generation, and are reported in the following.

Frequent itemset mining

This step aims at discovering frequent term sets from a transactional dataset in the form of frequent itemsets (see Definition 2.4.7, details in Section 2.3.1).

In the context of transactional dataset, a k-itemsets (i.e., an itemset of length k) is defined as a set of k distinct terms. Two itemsets are *disjoint* if they have no term in common. In addition, an itemset is said to *cover* a given transaction (i.e., sentence) tr_{jk} if all of its terms are contained in tr_{jk}. Consider an itemset I, a transactional representation T of a document collection D, the *support* of the itemset I in transactional dataset T is defined as ratio of the number of the transactions in T covered by I and the total number of transactions in T. An itemset I is said to be frequent in T, whose support value in $D \geq minsup$, where $minsup$ is a given minimum support threshold. For instance, the itemset {Data, Analysis} is a 2-itemset that covers 2 transactions in T, thus, its $support = \frac{2}{6}$. The GraphSum exploits Apriori algorithm [3] to accomplish the frequent itemset mining task. However, other itemset miners can also be integrated easily.

Term set correlation estimate

The evaluation of pair-wise correlations achieved in the previous step takes place at this step. To this goal, association rule mining [2] (see Section 2.4.1) technique is applied to discover and evaluate the pair-wise correlations among disjoint itemsets. An association rules (see Definition 2.4.1), in the format A⇒B, have been discovered from the given document collection. A number of quality measures are proposed to support the selection and ranking of the rules, which may be of interest [152]. Three foremost quality measures of association rules are (i) association rule support (see Definition 2.4.2), (ii) association rule confidence (see Definition 2.4.3), and (iii) association rule lift (see Definition 2.4.4). The itemsets A and B are not correlated of a rule A⇒B, if lift(A⇒B) = 1 or closer to 1 i.e., A and B are statistically independent. The lift(A⇒B) < 1 indicates negative correlation and lift(A⇒B) > 1 shows positive correlation i.e., the implication between itemsets A and B holds more than the expected one.

For example, consider the transactional dataset of document collection as reported in Table 3.2. The association rule {Data} ⇒ {Analysis} has a rule support equal to $\frac{2}{6}$ in T and the rule confidence equal to $\frac{2}{4}$, since the itemset {Data, Analysis}) appears twice in T, whilst the implication {Data} ⇒ {Analysis} holds in half of the transactions. The strength of the set terms associations measured in a rule confidence, however, may be misleading [153]. When the rule consequent with relatively high support value may be characterized with higher rule confidence, even if the rule has relatively lower strength in actual. To cope with this issue, *lift* index [152] may be applied instead of the rule confidence for measuring (symmetric) correlation among

the antecedent and consequent of the discovered rules.

To ensure high quality of the correlations among interested independent set of terms being marginal in the context of analyzed document collection, the GraphSum only considers frequent and highly correlated pair-wise correlations among the set of terms. In other words, the GraphSum summarizer exclusively selects associations rules, which satisfy the following criteria:

- rule support is equal to or exceeds a minimum support threshold *minsup*, and
- rule lift is in the range (0,$max^{-}lift$) or greater or equal to $min^{+}lift$, where $max^{-}lift$ and $min^{+}lift$ indicate the maximum negative and the minimum positive correlation thresholds.

The positive and negative correlations among the set of terms are differentiated during the summarization process. The task of generating association rules is usually accomplished firstly by generating then evaluating the all possible frequent itemsets [3]. To avoid generation of all possible association rules from the discovered frequent itemsets, the GraphSum adopts symmetry of lift measure [153] e.g., evaluation between term sets A and B in pair-wise correlation lift(A⇒B)=lift(B⇒A). The GraphSum summarizer takes into account distinct couple of disjoint frequent itemsets A and B exclusively such that the union of itemsets A and B i.e., A$\bigcup$B (the rule A⇒B) is frequent with respect to the minimum support threshold (*minsup*).

Correlation graph

The *correlation graph* is a graph-based model presenting the generated most significant and hidden correlations among set of terms. The *correlation graph* is formally described in Definition 3.3.2.

For example, consider three non-negative numbers $minsup = 1\%$, $max^{-}lift = \frac{4}{5}$, and $min^{+}lift = 10$. The support of {Data, Analysis} is 33%, and lift({Data}⇒{Analysis})=$\frac{3}{5}$ for the transactional dataset reported in Table 3.2. Then the corresponding *correlation graph* G comprises of two distinct nodes {Data} and {Analysis} (being frequent by Apriori principle [3] linked by an edge having weight=$\frac{3}{5}$ unlike previous approaches e.g., [94][170][175]) related to single terms. The nodes in the *correlation graph* may have the size of term set greater than one being a subset of a sentence e.g., {Information, Data}.

Definition 3.3.2 Correlation graph. *Let T be a transactional representation of a document collection D and three non-negative numbers i.e., minsup, max^{-}lift, and min^{+}lift. Let $\mathcal{I}$ be the set of frequent itemsets mined from T by enforcing minsup - a minimum support threshold. A correlation graph G built on T is a graph, whose nodes are frequent itemsets (term sets) in $\mathcal{I}$, whilst bidirected edges link the arbitrary node couples A and B, such that either lift(A,B) $\in$ (0,max^{-}lift) or lift(A,B) $\geq$ min^{+}lift. Edges are weighted by the corresponding rule lift.*

3.3.1.3 Graph indexing

A variant of PageRank [21] (a graph ranking algorithm) has been exploited in the GraphSum summarizer to measure the authoritative information of graph nodes because the term sets contained in the correlation graph are of not the same importance of the analyzed document collection. The PageRank algorithm [21] performs ranking graph nodes in such a way that it connects a graph node X with another node Y with an edge having some specific weight, analogous to a vote given from X to Y. The relatively importance of the node Y is higher in graph, if the sum of its weights incoming in Y is higher. The graph is modelled as Markov Chain Model, where probabilities of moving between the graph are described by Random Walks to address the indexing problem. The evaluated probabilities are obtained after an infinite walk on the graph in a stationary state of Random Walk. The graph nodes represent term sets in the context of analyzed document collection, thus, the stationary probabilities reflect the expected probabilities of occurrences of each term set. An iterative algorithm (in PageRank algorithm) is exploited, which aggregates transitional probabilities between all graph nodes till the steady state is reached for estimating the authority of a graph node. The resultant ranking of the graph node is called PageRank score based on the authority score.

The PageRank score $PR(N_i)$ of a graph node N_i formally could be approximated as in the following Formula:

$$PR(N_i) = (1 - d) + d \cdot \sum_{k=1}^{n} \frac{PR(N_k)}{C(N_k)} \tag{3.1}$$

where n is the number of edges incoming into N_i, $PR(N_k)$ is the PageRank score of an arbitrary N_i's neighbor N_k, $C(N_k)$ is the outgoing degree of the node N_k, and $d \in [0, 1]$ is a damping factor that weights PageRank score propagation from one node to another that is often set to 0.85 [21].

To distinguish the contribution of positive and negative correlations of term sets in graph ranking, a variant of traditional PageRank score has been exploited in the GraphSum given in Formula 3.2.

$$PR(N_i) = (1-d) + d \cdot (\sum_{k=1}^{n} \frac{1}{\sqrt{C^{-}(N_k)}} \frac{PR(N_k)}{C(N_k)}) \tag{3.2}$$

where $C^{-}(N_k)$ is the outgoing degree of a node N_k exclusively computed by considering negatively correlated edges. This idea penalizes the PageRank score based on the contribution of negatively correlated terms. The penalization factor is $\frac{1}{\sqrt{C^{-}(N_k)}}$, which mitigates the propagation of PageRank score from each node N_k to N_i with a significant number of negatively correlated neighbours. Theoretically, penalizing score corresponds to re-scaling of the transition matrix terms related to negatively correlated pair of nodes, which flattens the probability of randomly choosing a negatively correlated link to follow in a Random Walk. The nodes that are exclusively linked to their neighbourhood through positively correlated links have penalization zero.

3.3.1.4 Sentence selection

The GraphSum adopts graph indexing approach to evaluate and select the most relevant and informative sentences for including in the summary. In particular, two key steps are considered (i) relevance score based on PageRank indexes, and (ii) correlation graph coverage.

Sentence relevance score

The relevance score of a sentence s_{jk} belonging to the analyzed document collection measures its significance in terms of authority of its contained nodes (term sets) in the correlation graph. The relevance score of a sentence s_{jk} can be defined as the normalized sum of the PageRank scores assigned to each node (i.e., term set) contained in s_{jk}, mathematically computed as given in the following formula:

$$SR(s_{jk}) = \frac{\sum_{i \mid n_i \subseteq t_{jk}} PR_{ik}}{N_{t_{jk}}} \tag{3.3}$$

where t_{jk} is the transaction related to s_{jk} sentence, $\sum_{i \mid n_i \in t_{jk}} PR_{ik}$ is the sum of the PageRank scores PR_{ik} associated with every node n_i covering t_{jk} in the correlation graph, and $N_{t_{jk}}$ is the total number of nodes covering t_{jk}.

Sentence coverage

The pertinence of a sentence s_{jk} with respect to the correlation graph G is measured by sentence coverage. Lets consider, each sentence $s_{jk} \in D$ a binary vector represented as *sentence coverage vector* (SC_{jk}):

$$SC_{jk} = \{sc_1, \ldots, sc_{|N|}\} \tag{3.4}$$

where $|N|$ is the number of nodes contained in the correlation graph G and $sc_i = \mathbf{1}_{tr_{jk}}(n_i)$ indicating whether a term set n_i covers or does not cover tr_{jk}. $\mathbf{1}_{tr_{jk}}$ is an indicator function given an arbitrary term set n_i contained in G and is formally defined in the following:

$$\mathbf{1}_{tr_{jk}}(n_i) = \begin{cases} 1 & \text{if } n_i \subseteq tr_{jk}, \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

The coverage of a sentence s_{jk} with respect to the correlation graph is defined as the number of ones contained in the corresponding coverage vector SC_{jk} (see Equation 3.4).

The problem of selecting the most informative and representative sentences from the analyzed document collection in terms of coverage and relevance score is formalized as set covering problem described in the following.

The set covering problem The focus of set covering problem is on selection of minimal set of sentences with arbitrary size l and maximal score, whose logic OR operation of corresponding coverage vectors (i.e., $SC^*{=}SC_1 \vee \ldots \vee SC_l$) generates a binary vector with the maximum number of ones.

The set covering optimization problem focuses on selecting the minimal set of sentences, whose corresponding coverage vectors (i.e., $SC^*{=}SC_1 \vee \ldots \vee SC_l$) generates a binary vector with the maximum ones. The SC^* vector is denoted as the *summary coverage vector* in the subsequent sections. The set coverage problem is addressed by the GraphSum for selection of sentences with maximal model coverage and relevance score to include in the summary.

The set covering optimization problem is computationally difficult and NP-hard (i.e., non-deterministic polynomial-time hard) to tackle. Therefore, a greedy strategy similar to the one successfully applied in [12] has been exploited in the context of document summarization in the GraphSum . The greedy strategy considers sentences that cover the maximum number of graph nodes for the sentence selection. Thus, it is the most fascinating feature.

Algorithm 5 Greedy strategy

Require: set of sentences S, set of sentence coverage vectors SC, and set of sentence relevance scores SR
Ensure: summary $\mathcal{SU}$
1: $\mathcal{SU} = \emptyset$
2: SC^* = set_to_all_zeros() {initialize the summary coverage vector with only zeros}
{Cycle until SC^* contains only 1s (i.e., until the generated summary covers all the itemsets of the model) }
3: **while** SC^* contains at least one zero **do**
4: $MaxOnesSentences$ = max_ones_sentences(S, SC) {Select the sentences with the highest number of ones}
5: **if** $MaxOnesSentences$ is not empty **then**
6: $s_{best} = arg\,max_{s_j \in MaxOnesSentences} SR(s_j)$ {Select the sentence with maximum relevance score among the ones in $MaxOnesSentences$}
7: $\mathcal{SU} = \mathcal{SU} \cup s_{best}$ {Add the best sentence to the summary}
{Update the summary coverage vector SC^*. $SC_{s_{best}} \in SC$ is the sentence coverage vector associated with the best sentence s_{best} }
8: $SC^* = SC^*$ OR $SC_{s_{best}}$ { Set the bits associated with the term sets covered by s_{best} to one}
{Update the sentence coverage vectors in SC}
9: **for all** SC_i in SC **do**
10: $SC_i = SC_i$ AND $\overline{SC^*}$ {Set the bits of SC_i associated with the term sets already covered by the summary to zero}
11: **end for**
12: **end if**
13: **end while**
14: **return** $\mathcal{SU}$

The sentences with equal terms and the highest coverage characterized by its highest relevance score SR are preferred. The detailed description of the greedy strategy is reported in the following.

The greedy strategy The applied sentence selection algorithm identifies sentence s_{jk} with the best complementary vector SC_{jk} with respect to the current summary coverage vector SC^* at every step. In other words, the algorithm finds the sentence s_{jk}, which covers the maximum number of graph nodes that are not covered by any sentence already contained in the current summary.

A pseudo-code of the greedy strategy for selecting sentences is given in Algorithm 5. It takes three inputs set of sentences S, set of sentence coverage vectors SC, and set of sentence relevance scores SR. The algorithm produces a summary $\mathcal{SU}$ composed of the sentences, which cover the correlation graph in the best way.

The step-wise procedure for the sentence selection involves the following steps. Firstly, initializing variables (lines 1-2). Next, insertion of the best sentence into the summary is iteratively carried out (lines 3-13). At every iteration, sentences with maximum coverage i.e., their coverage vector comprises of maximum number of ones (line 4), sentence having maximal relevance score (Cf. Formula 3.3) is preferred (line 6). Finally, the selected

sentence s_{best} is included in the summary $\mathcal{SU}$ (line 7). The summary and sentence coverage vectors are updated accordingly (lines 8-11). The updating process excludes set of coverable graph nodes from the ones, which already have been covered by the current summary. The procedure iterates until the graph-based model is fully covered by the summary i.e., until the summary coverage vector contains only ones (line 3). Note that when $minsup>0$, each frequent itemset contained in the model covers at least one document sentence. Thus, the sentence selection process always succeeds in fully covering the graph.

3.3.2 Experimental results

A number of experiments has been carried out to ensure the quality of the GraphSum and to address the issues such as (i) A performance comparison between the GraphSum and its competitors on benchmark document collection, and (ii) an evaluation of the effectiveness of the GraphSum summarizer on real-world news document collections, and (iii) an analysis of the impact of the model parameters and features on the GraphSum performance. In the following section, a brief description is presented about considered data collection, and then subsequent sections describe the achieved results for the GraphSum summarizer.

3.3.2.1 Document collections

The performance of the GraphSum summarizer is tested on (i) Benchmark collections of English-written documents i.e., the task 2 datasets of DUC'04 [44]. These documents being the benchmark collections are used for the Document Understanding Conference (DUC) on multi-document summarization and (ii) real-world document collections, i.e., five English-written set of news articles.

The latest DUC dataset on generic English-written multi-document summarization contains task 2 datasets of DUC'04 [44] among the 50 document groups available in DUC'04 documents. Each one includes 10 English-written documents and (at least) one golden summary for each of the DUC'04 document groups. Five English-written news articles retrieved from Web in the time period of August 2011 to November 2011 has been also tested by the GraphSum . Each real-world set of news articles is related with a different topic and comprises of 10 news articles. Documents within each collection

cover the same topical subject, particularly, the retrieved news documents cover following topics:

- **Italian austerity**: The package of austerity measures approved by the Italian Government to lead Italy out of its debt crisis.
- **World terrorism**: The war against terrorism of the U.S.A. government
- **Strauss Kahn scandal**: Dominique Strauss Kahn charged with sexual assault
- **Lybia war**: The civil war breaks out in the North African state of Libya
- **Irene hurricane**: The Irene Hurricane strikes down on the U.S. East Coast

Each collection is made availble by querying interested topics in the Google News search engine, then selecting the top-10 news articles. The interested topics represents the different case studies, which are (i) very focused ones interesting for a short time period (e.g., Strauss Kahn), (ii) averagely focused ones that impact on future events (e.g., Irene Hurricane), and (iii) broader-spectrum and multi-faceted news articles (e.g., World Terrorism).

3.3.2.2 DUC'04 Benchmark documents summarization

The performance of the GraphSum has been analyzed on DUC'04 document collections. The comparison has been carried out with (i) all 35 summarizers submitted to the DUC'04 conference, (ii) the 8 human generated summaries provided by the DUC'04 system, (iii) summaries generated by two widely used open-source text summarizers Open Text Summarizer (OTS) [129] and TexLexAn [155], and (iv) a recently proposed itemset-based summarizer ItemSum [12]. The summaries of DUC'04 competitors are available in DUC'04 system [44], whilst summaries of other competitors (i.e., OTS, TexLexAn, and ItemSum) are achieved exploiting the algorithm configurations suggested by corresponding authors. For instance, the best configuration of itemSum reported in [12] is $minsup$=3% a minimum support threshold and ms=12 as model size. The standard configuration[1] of the GraphSum summarizer is the following minimum support threshold ($minsup$) equals to 3%, the maximum negative correlation threshold ($max^{-}lift$) equals to 0.6, and the minimum positive correlation threshold ($min^{+}lift$) equals to 15.

The GraphSum performance has been evaluated using ROUGE toolkit [91], which is adopted as official DUC'04 tool for performance evaluation[2].

[1] GraphSum standard configuration: $minsup$=3%, $max^{-}lift$=0.6, $min^{+}lift$=15

[2] The provided command is: ROUGE-1.5.5.pl -e data -x -m -2 4 -u -c 95 -r 1000 -n 4 -f A -p 0.5 -t 0 -d -a

Table 3.3: DUC'04 Benchmark documents. Comparisons between GraphSum and other competitors. Statistically relevant differences (with GraphSum standard configuration) and other approaches are starred.

Summarizer		ROUGE-2			ROUGE-SU4		
		R	Pr	F	R	Pr	F
TOP RANKED DUC'04 PEERS	peer67	0.089*	0.095*	0.092*	**0.015**	0.017*	0.016*
	peer120	0.076*	0.103*	0.086*	**0.015**	0.018*	0.016*
	peer65	0.087*	0.091*	0.089*	**0.015**	0.016*	0.015*
	peer66	0.086*	0.093*	0.089*	0.013	0.014*	0.014*
	peer121	0.071*	0.085*	0.077*	0.012*	0.014*	0.013*
	peer11	0.070*	0.087*	0.077*	0.012*	0.015*	0.012*
	peer44	0.075*	0.080*	0.078*	0.012*	0.013*	0.012*
	peer81	0.077*	0.080*	0.078*	0.012*	0.012*	0.012*
	peer124	0.078*	0.082*	0.080*	0.011*	0.012*	0.011*
	peer35	0.081*	0.085	0.083*	0.010*	0.011*	0.011*
DUC'04 HUMANS	A	0.088*	0.092*	0.090*	0.009*	0.010*	0.010*
	B	0.091	0.096	0.093*	0.013	0.013	0.013*
	C	0.094	0.102	0.098	0.011*	0.012*	0.012*
	D	0.100	**0.106**	0.102	0.010*	0.010*	0.010*
	E	0.094	0.099	0.097	0.011*	0.012	0.012*
	F	0.086*	0.090	0.088*	0.008*	0.009*	0.009*
	G	0.082*	0.087*	0.084*	0.008*	0.008*	0.007*
	H	**0.101**	0.105	**0.103**	0.012*	0.013*	0.012*
OTS		0.069*	0.079*	0.074*	0.008*	0.009*	0.009*
TexLexAn		0.059*	0.068*	0.063*	0.006*	0.007*	0.007*
itemSum		0.083*	0.085*	0.084*	0.012*	0.014*	0.014*
GraphSum		**0.093**	**0.099**	**0.097**	**0.015**	**0.021**	**0.019**

The quality of summary is measured by counting unit overlaps between candidate summary and a set of reference summaries (i.e., golden summaries) available in DUC'04 [44]. The summarizer achieving the highest ROUGE scores could be declared the most effective one among others. To ensure fair comparison among the competitors, the summaries have been preliminary normalized by truncating their size to 665 bytes before using ROUGE toolkit. A number of evaluation scores are implemented in ROUGE toolkit [91]. However, in the performance evaluation of the GraphSum , only two ROUGE evaluator scores (i.e., ROUGE-2 and ROUGE-SU4) are reported, since similar results have been obtained in other ROUGE evaluators.

The ROUGE evaluation results are reported in Table 3.3 on DUC'04 benchmark documents containing 10 most effective competitors of DUC'04, 8 human-generated summaries, OTS, TexLexAn, itemSum and GraphSum summarizers. All three available versions of the top ranked summarizer of DUC'04 (i.e., CLASSY [35]) are reported. To ensure validation for statistically significance of the GraphSum performance against its available competitors, a paired t-test [41] at 95% significance level has been exploited for all evaluated measures. The GraphSum summarizer significantly outperforms as compared to its competitors OTS, TexLexAn, and ItemSum for all

tested measures. Additionally, it performed pretty better than all the available DUC'04 competitors based on ROUGE-2 and ROUGE-SU4 F1-measure. The performance of the GraphSum is observed as good as the most effective competitors (i.e., CLASSY [35] and Peer120) on the basis of ROUGE-SU4 Recall. However, CLASSY exploits a semantics-based preprocessing step. Differently, the GraphSum is a general-purpose summarizer that does not integrate any advanced linguistic analysis.

In some cases, the summarizers that outperform the human-generated summaries are the only GraphSum and CLASSY. For example, both GraphSum and peer67 outperformed all the human-generated summaries in terms of ROUGE-SU4 F1-measure. More specifically, the GraphSum significantly outperformed 4 out of 8 human-generated summaries in terms of ROUGE-2 F1-measure, where as peer67 outperformed 2 out of 8 summaries.

3.3.2.3 Real-world news articles summarization

The summaries achieved by the GraphSum summarizer and its competitors, whose algorithm implementation is publicly available on the real-world news articles and their performance evaluation using ROUGE toolkit [17] are reported in this section.

Summary comparison

The summaries generated by the GraphSum and its tested competitors (i.e., TexLexAn, OTS, itemSum) on Irene hurricane topic of news articles collection are reported in Table 3.4 as representatives of other collections. The configuration suggested by corresponding authors of the tested competitors has been adopted to generate the summaries.

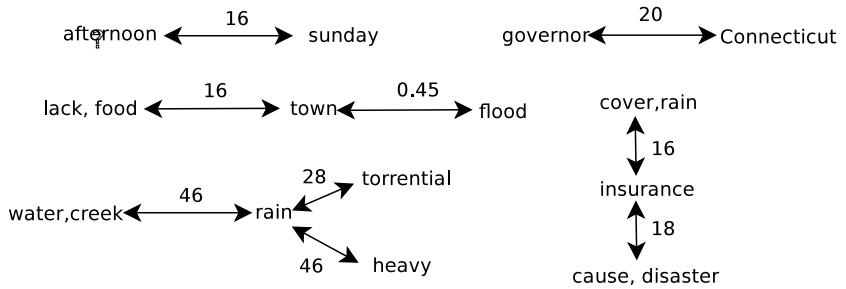


Figure 3.3: Portion of the extracted correlation graph. Irene Hurricane news articles

Table 3.4: Summary examples. Irene hurricane news articles

Method	Summary
GraphSum	New York was pounded by heavy winds and torrential rain on Sunday morning as Hurricane Irene bore down on the city, threatening to cause flash flooding and widespread damage in the US's most populous city. It's one of several towns in states such as New Jersey, Connecticut, New York, Vermont and Massachusetts dealing with the damage of torrential rain and flooding spawned by Hurricane Irene.
itemSum	New York was pounded by heavy winds and torrential rain on Sunday morning as Hurricane Irene bore down on the city, threatening to cause flash flooding and widespread damage in the US's most populous city. "If this is what it means to live in the nanny state, I'm very content," Krasnow said.
OTS	As emergency airlift operations brought ready-to-eat meals and water to Vermont residents left isolated and desperate, states along the Eastern Seaboard continued to be battered Tuesday by the after effects of Irene, the destructive hurricane turned tropical storm. Dangerously-damaged infrastructure, 2.5 million people without power and thousands of water-logged homes and businesses continued to overshadow the lives of residents and officials from North Carolina through New England, where the storm has been blamed for at least 44 deaths in 13 states.
TexLexAn	As emergency airlift operations brought ready-to-eat meals and water to Vermont residents left isolated and desperate, states along the Eastern Seaboard continued to be battered Tuesday by the after effects of Irene, the destructive hurricane turned tropical storm. Search-and-rescue teams in Paterson have pulled nearly 600 people from flooded homes in the town after the Passaic River rose more than 13 feet above flood stage, the highest level since 1903.

The GraphSum generated the most focused summary providing concise yet comprehensive insight of the analyzed news. Portion of the extracted correlation graph from Irene hurricane news articles is reported in Fig. 3.3. As an example, three pair-wise correlations of term set with relatively high lift (i.e., lift($\{Rain\}$, $\{Heavy\}$) = 46) extracted from the considered news articles are $\{Rain\} \leftrightarrow \{Torrential\}$, $\{Rain\} \leftrightarrow \{Heavy\}$, and $\{Rain\} \leftrightarrow \{Water,Creek\}$. These correlations are highly significant, since these appear within the top-2 sentences of the summary. Whilst, $\{Town\} \leftrightarrow \{Flood\}$ correlation is not selected by the GraphSum because of its negative correlation (lift=0.45). The same correlation is selected and included in the summary by TexLexAn.

Unlike GraphSum summary, the summaries produced by other competitors seem to be generic ones and contain partially redundant terms. For example, ItemSum generated summary containing the sentence that entails marginal information regarding the interested topical news *"If this is what it means to live in the nanny state, I'm very content" Krasnow said.*

Performance comparison

The performance of the GraphSum on real-world news articles has been compared with its competitors (i.e., TexLexAn [155], OTS [129], itemSum [12] since their source is publicly available) using the ROUGE tooklit [91].

Table 3.5: News articles. Comparisons between GraphSum (with standard configuration) and other approaches. Statistically relevant differences between GraphSum and other approaches are starred.

Article	Summarizer	ROUGE-2			ROUGE-SU4		
		R	Pr	F	R	Pr	F
ITALIAN AUSTERITY	OTS	**0.044**	**0.313**	**0.077**	0.014*	0.101*	0.024*
	TexLexAn	0.039	0.283*	0.068	0.009*	0.068*	0.016*
	itemSum	0.038	0.265*	0.067*	0.009*	0.065*	0.016*
	GraphSum	0.042	0.299	0.073	**0.015**	**0.108**	**0.027**
WORLD TERRORISM	OTS	0.007*	0.069*	0.013*	0.001*	**0.002**	0.002*
	TexLexAn	0.008	0.073*	0.015	0.001*	0.001*	0.001*
	itemSum	0.008	**0.118**	0.015	0.002*	0.001*	0.002*
	GraphSum	**0.010**	0.085	**0.017**	**0.004**	**0.003**	**0.005**
STRAUSS KAHN SCANDAL	OTS	0.017*	0.146*	0.030*	0.002*	0.015*	0.003*
	TexLexAn	0.018*	0.162*	0.032*	0.002*	0.014*	0.003*
	itemSum	0.019*	0.192*	0.035*	0.002*	0.019*	0.003*
	GraphSum	**0.023**	**0.198**	**0.040**	**0.004**	**0.040**	**0.008**
LIBYA WAR	OTS	**0.012**	0.134	**0.022**	0.001*	0.002*	0.001*
	TexLexAn	**0.012**	**0.138**	**0.022**	0.001*	0.001*	0.001*
	itemSum	0.005*	0.114	0.009*	0.002*	0.002*	0.001*
	GraphSum	**0.012**	0.135	**0.022**	**0.004**	**0.004**	**0.004**
IRENE HURRICANE	OTS	0.011*	0.108*	0.021*	0.002*	0.001*	0.002*
	TexLexAn	0.012*	0.122*	0.023*	0.001*	0.002*	0.002*
	itemSum	0.006*	0.153	0.012*	0.002*	0.002*	0.002*
	GraphSum	**0.016**	**0.157**	**0.029**	**0.005**	**0.005**	**0.006**

Due to unavailability of golden summaries for the real-wolrd news articles, the performance comparison between the GraphSum and the other considered competitors has been carried out using the leave-one-out cross validation approach done in [34]. In particular, nine out of ten news documents have been summarized leaving the one (not considered) for each news category. The generated summary discovered from nine documents is compared with the remaining one, which was not considered and holds position as golden summary for the news category. Furthermore, the tests are carried by varying the golden summary and computing average performance scores precision (P), Recall (R), and F1-measure (F1) for ROUGE-2 and ROUGE-SU4 evaluators of ROUGE toolkit by all the considered summarizers (i.e., GraphSum , itemSum, OTS, TexLexAn). Consideration of one document as golden summary in the news articles context is a good quality approximation, since all the document of a category address the same interested topical news.

Table 3.5 reports obtained ROUGE scores for the summarizers on news articles collection. Again, to validate the statistical significance of the GraphSum performance, a paired t-test [41] at 95% significance level is applied for all evaluated measures. Statistically relevant differences in the comparisons are distinguished with stars* between the GraphSum and the other summarizers. Besides, the most effective results of summarizer(s) are highlighted with boldface.

The promising results on the basis of ROUGE scores are achieved in the context of real-world news articles. The GraphSum summarizer outperforms than that of the other considered summarizers in terms of ROUGE scores ROUGE-SU4 Precision, Recall, and F1-measure. For instance, the GraphSum produces better results in 4 out of 5 news article collections with respect to ROUGE-2 F1-measure.

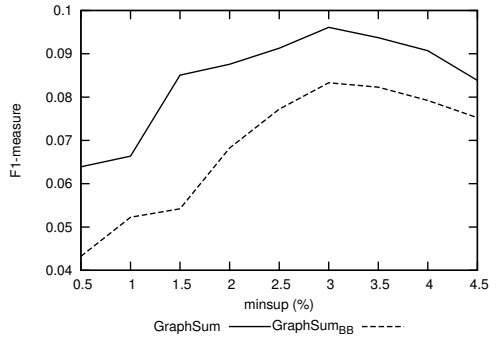
3.3.2.4 Performance analysis

The impact of parameters and features is analyzed this section for the GraphSum summarization performance, such as (i) input parameters, (ii) greedy approach usage for graph-based model coverage, and (iii) choice of type of extracted itemsets. These factors are described in the following.

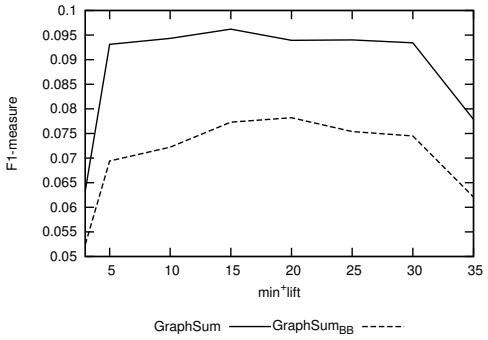
Impact of input parameters

The varying the input parameters (i.e., *minsup* and *minlift*) of the GraphSum summarizer, the obtained ROUGE-2 F1-measure scores being as representatives of all achieved measures are plotted in Fig. 3.4. The affect of the support threshold is pretty significant on the generated summary. The high support threshold (e.g., 7%) when enforced, discarded great number of potentially significant patterns. Thus, potentially relevant patterns remained undiscovered due to high support threshold, and the selected sentences eventually are composed of relatively less informative term sets. Instead, very low support values (e.g., 0.5%) may be outfitted, hence, the generated model coverage pruned to errors. Whilst, medium support threshold values (e.g., 3%) presents the best-trade-off between model, and general purpose results, therefore, averagely accurate summaries have been generated.

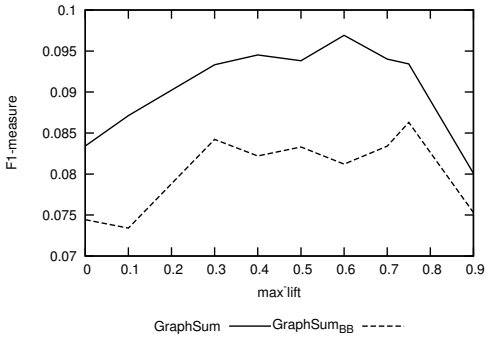
The GraphSum performance is slightly affected when the maximum negative lift ranges in [0.4,0.75] and the maximum positive lift ranges in [5,25]. The performance decreases if lift threshold values are configured out of these ranges. Especially, lift closer to 1 may prejudice the quality of the



(a) Impact of *minsup*. $max^{-}lift$=0.6. $min^{+}lift$=15.



(b) Impact of $min^{+}lift$. *minsup*=3%. $max^{-}lift$=0.6.



(c) Impact of $max^{-}lift$. *minsup*=3%. $min^{+}lift$=15.

Figure 3.4: Parameter analysis and comparison between GraphSum and GraphSum $_{BB}$ in terms of Rouge-2 F1-measure

generated model, where as increasing further the lift threshold values (i.e., max^{-}lift<0.4 or min^{+}lift>25) may prune the informative patterns suitable for summarization.

Choice of coverage strategy

The set coverage optimization problem has been addressed using greedy strategy in the GraphSum summarizer, which in general, selects the sentences that cover the graph-based model in the best manner. The set covering problem being a min-max could be managed as linear programming problem and combinatorial optimization strategies may be adopted. Therefore, the performance evaluation has been compared of the GraphSum and its slightly modified version (called GraphSum $_{BB}$) to analyze the impact of greedy algorithm on summarization process. The GraphSum $_{BB}$ adopts a

branch-and-bound algorithm [122] to address the set covering problem. The comparison between the results of the GraphSum and GraphSum $_{BB}$ in term of ROUGE-2 F1-measure is shown in Fig. 3.4 by varying *minsup* and *lift* values. The GraphSum summarizer outperformed GraphSum $_{BB}$ with all the analyzed configurations. This behaviour of the GraphSum indicates the more stability and low sensitivity of the generated model towards errors as well as data over-fitting. Moreover, the GraphSum acquires lesser execution time with respect to GraphSum $_{BB}$, for instance, at least 20% lesser time in all considered settings.

Choice of type of extracted itemsets

The other two variants of the GraphSum summarizer have also been analyzed, which considered maximal [126] and frequent closed itemsets [118] mining algorithms instead of frequent itemsets. Results of the variants, unfortunately, worsen the GraphSum performance averagely. This behaviour indicates the fact that only subset of entire frequent itemsets is considered and thus, potentially essential correlations are not considered during the sentence selection.

3.4 Analysis and visualization of user-generated content

Recently, social networks and online communities are becoming a popular way of exchanging data. Significant efforts are made for analysis of textual data published on social networks. For example, analysis of tweets online published on Twitter micro-blogging website produced potential results in user behaviour profiling [89] [101] and topic trend discovery [32]. Although the quality and correctness of the contents still remains questionable, yet micro-blogging and user tweets provide information about incidents taken place in a very short time.

In the context of textual data analysis, a widely exploratory data mining technique association rule mining [2] (see Section 2.4.1) allows the discovery of correlations of terms (or set of terms) amongst the analyzed data. Although association rule mining algorithms have some limitations in terms of support threshold (see Definition 2.4.2) that may not extract the relationship of terms appeared infrequent in the analyzed data but possess significant

impact on the knowledge discovery. To overcome this issue, [143] firstly proposed the discovery of generalized association rules. The investigation of textual data in different application context exploiting generalized association rule mining has been already addressed. Generalized association rule (see Definition 3.4.3) may contain high level (i.e., generalized) concepts (or terms). Thus, it represents underlying correlations at different abstraction levels. A taxonomy (i.e., a set of is-a hierarchies) may be exploited to aggregate the analyzed textual data terms into higher level concepts, which are less likely to be infrequent for the analyzed textual data.

Visual tools has been proposed to support analysts in knowledge discovery process focusing on either visualizing association rule mining results to ease experts in validation tasks [88] [168] [103] or allowing experts to manually drive the data mining process [52] [90]. However, visualization of generalized rules mined from textual data published on social networks has never been investigated so far. To cope this issue, the proposed visualizer tool is reported in the following sections.

3.4.1 Twitter Generalized Rule Visualizer (TGRV)

A novel visualization tool called *Twitter Generalized Rule Visualizer (TGRV)* has been developed allowing experts to explore the results of generalized rule mining process from textual data coming from social network micro-blogging website Twitter[3] (i.e., tweets or user-generated content) effectively. Tweets over the same topic are initially retrieved through Application Programming Interfaces (APIs), and then are integrated into a common textual documents. Furthermore exploiting a WordNet taxonomy built over document terms, frequent generalized association rules are discovered from the generated dataset of tweet's textual documents. In the end, a graph-based visualization model - *Generalized Rule Graph* represents the high level association among the terms present in the considered textual documents. Generalized Rule Graph allows experts to explore the detected generalized rules from different perspectives and at different abstraction levels. The graph nodes in the visualization model represent term sets of arbitrary size, whilst edges represent strong associations among pair of nodes. The provision of different viewpoints of the TGRV may help analysts to avoid exploring the entire set of frequent rules. For instance, change in abstraction level in rule analysis allows experts to tackle the limitations of traditional rule visualizers that usually provide unsatisfactory results while dealing with sparse dataset.

[3] http://www.twitter.com

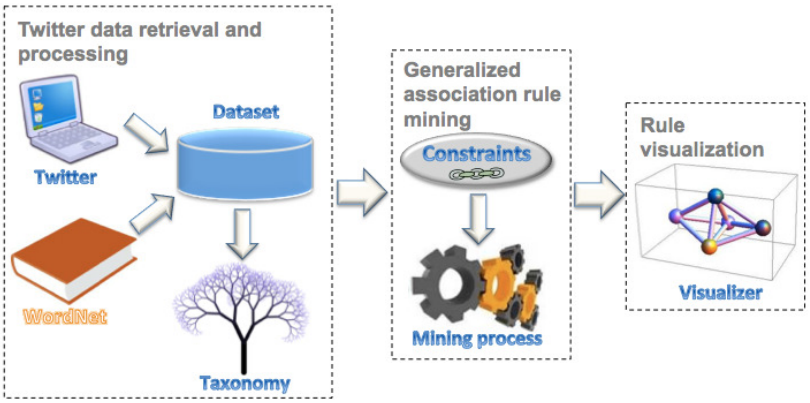


Figure 3.5: The TGRV system architecture

In Fig. 3.5, the system architecture is illustrated and main blocks of the tool are briefly described in the following.

3.4.1.1 Twitter data retrieval and processing

The retrieval of the tweets posted on social networking website (i.e, Twitter) comprising of at most 140 characters long and publicly visible is carried out by means of general purpose tool - Application Programming Interfaces (APIs), which allows efficient retrieval of the social network data. However, the received tweets are unsuitable for the mining processes. Besides, differentiation of tweets between closed time-interval (i.e., during last 12 hours) form the ones that were published long time in past (i.e., tweets published the day before). TGRV Tool crawls tweets published on a specific topic by configuring a number of filtering parameters such as selection of keywords, geographical radius are used for selection of interested tweets from the Public stream of tweets.

Definition 3.4.1 Transactional Twitter dataset. *Let TW be a set of tweets, the transactional Twitter dataset T associated with TW is a set of transactions T_i, one for each tweet such that $tw_i \in T$. Each transaction T_i=$\{t_1, t_2, \ldots, t_n\}$ be a set of terms (stems) relative to tw_i.*

The collected tweets being unsuitable for the further steps are then applied an ad-hoc preprocessing phase that includes data cleaning and processing steps. Specially, numbers, stopwords, links, non-ascii characters, and replies are eliminated from the textual tweet contents. Then, the traditional

stemming algorithm based on WordNet [169] is applied to make tweets transformation into their root form (i.e., stems). Each tweets is mapped into stems and each one corresponds to a distinct stem named as transactional twitter dataset (see Definition 3.4.1).

For example, consider the tweet: *"One of the oldest transportation infrastructures is in New York!"*, which may be mapped into corresponding transaction {*One, old, transportation, infrastructure, New York City*} comprising of 5 distinct terms (i.e., stems).

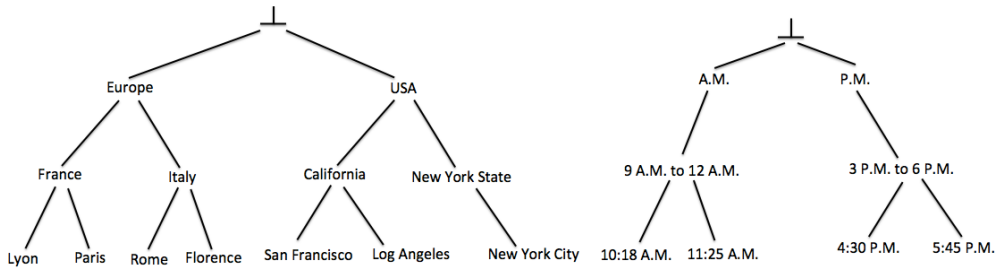


Figure 3.6: Examples of generalization hierarchies

The generalized association rule detection is enabled by building a taxonomy θ over transactional Twitter dataset T. A taxonomy θ is a set of generalization hierarchies, where each generalization hierarchy is tree-based representation with leaf nodes belonging to term in transactional Twitter dataset T. Whilst higher (i.e., upper) level nodes referred to as *generalized terms* are aggregated higher level concepts of T (i.e., leaf nodes). The level of a generalized term tr with respect to θ is defined as the height of the θ's subtree rooted in tr.

The two generalization hierarchies, which aggregate terms into relative higher concepts of specific cities and points of time are depicted in Fig. 3.6. The higher level of concepts of Europe, USA, France and Rome are 3, 3, 2 and 1 respectively. likewise, other hierarchies may be aggregated having different semantics such as animals may be classified according to their specie and class. The TRGV adopts semi-automatic and analyst-driven approach to generate a taxonomy over the transactional dataset achieved from textual data contents. To build taxonomy of generalization hierarchies over transactional Twitter dataset T, the WordNet lexical database [169] is exploited to obtain the most relevant semantic relationships among tweet terms. In particular, TGRV utilizes the hypernyms (is-a -part-of relationships) or hyponyms (i.e., is-a-subtype-of relationships) to generate taxonomy of the tweet terms. For instance, the term New York City may be generalized as New York

State, since the semantic relationship <New York City> is-a-part-of <New York State> is retrievable from the WordNet database. Furthermore, the deep generalization level may allow aggregating/mapping New York City as USA, because <New York City> is-a-subtype-of <U.S.A.>. Generally, a taxonomy includes items having several generalization hierarchies.

3.4.1.2 Generalized association rule mining

A two-step process accomplishes the generalized association rule mining [143] (i) Frequent generalized itemset mining driven by a minimum support threshold, and (ii) generalized association rule generation starting from the previously extracted generalized itemset.

Frequent generalized itemset mining

While dealing with textual data terms enriched with their generalized higher level concepts, a generalized *k-itemset* is defined as a set of k distinct terms or generalized terms. For example, referring to the taxonomy depicted in Fig. 3.6 and tweets transactional dataset (i.e, {*One, old, transportation, infrastructure, New York City*}), the {*transportation, U.S.A.*} is an example of generalized itemset of 2-length (i.e., 2 items). Notice that itemsets are special cases of generalized itemsets which exclusively include not generalized terms.

The itemset mining has been derived by means of a well-known support quality index [2]. To understand the generalized itemset support, the concept of generalized itemset matching (see Definition 3.4.2) is described.

Definition 3.4.2 Generalized itemset matching. *Let T be a transactional Twitter dataset and θ be a taxonomy built over term in T. A generalized itemset X matches an arbitrary transaction t in T, if and only if for each (possibly generalized) term tr in X:*

- *tr is a leaf node of θ and is contained in t or*
- *tr is a non-leaf node of T and there exists a descendant dtr of tr with respect to θ such that dtr is contained in t.*

Therefore, the observed frequency in the analyzed dataset correlates with the support of a non-generalized itemset. A generalized itemset is said to be frequent if its support is equal to or higher than a given minimum support threshold *minsup*.

Generalized association rule extraction

Generalized association rules are implications among set of terms, possibly include generalized terms. A more formal definition is given in the following.

Definition 3.4.3 (Generalized association rule. *Let A and B be two generalized itemsets such that $A \bigcap B = \phi$. A generalized association rule is represented in the form $A \Rightarrow B$, where A and B are the body (antecedent) and the head (consequent) of the rule respectively.*

For example, the generalized rule $\{NewYorkState\} \Rightarrow \{transportation\}$ represents a co-occurrence relationship among two generalized itemsets of 1-length [143]. The generalized association rule discovery is usually driven by minimum rule support (*minsup*) and confidence (*minconf*) thresholds [143], described as follow.

Definition 3.4.4 Generalized association rule support. *Let D be a dataset comprises of transactional twitter dataset T and taxonomy θ with generalized hierarchies of terms in T. Let a generalized association rule be $A \Rightarrow B$ such that $\{A, B\} \epsilon D$. The support $Support(A \Rightarrow B)$ is defined as the frequency of itemsets $A \bigcup B \epsilon D$.*

The support generally represents the prior probability of $A \bigcup B$ (i.e., its observed frequency in the analyzed textual data).

Definition 3.4.5 Generalized association rule confidence. *Let D be a dataset comprises of transactional twitter dataset T and taxonomy θ with generalized hierarchies of term in T. Let a generalized association rule be $A \Rightarrow B$ such that $\{A, B\} \epsilon D$ and $Support(A \bigcup B)$ be the frequency of $A \bigcup B \epsilon D$. Its confidence is defined as $\frac{Support(A \bigcup B)}{Support(A)}$*

For example, consider $\{NewYorkState\} \Rightarrow \{transportation\}$ be a generalized rule at *(minsup=10%, minconf=50%)* meaning that the term *New York State* and *transportation* co-occur in 10% of the analyzed data (at different abstraction levels). The confidence 50% indicates *transportation* occurred half of the times where term *New York State* occurred. Hence, it is an estimation of the strength of the term implications. TGRV Tool entails discovering all frequent generalized itemsets from a given transactional Twitter dataset T, when provided by a taxonomy θ built over T, and a minimum support threshold *(minsup)* and minimum confidence threshold *(minconf)*.

3.4.1.3 Rule visualization

The proposed TRGV Tool - a novel graph-based visualization model allows analysts to easily explore the generalized rule mining results from different viewpoints. Moreover, since rules are detected from large textual sparse datasets of high dimensionality, therefore, a textual representation of mined patterns may neither be suitable for inspection manually nor easily interpretable. Analysts, in the context of generalized association rule mining, may not be only interested in a simple representation of each single rule, instead they may look for relationships between rules at their different abstraction levels.

The representation of TRGV Tool consists of a graph - *Generalized Rule Graph*, in which nodes represent frequent generalized itemsets, while edges represent associations between the nodes. A more formal definition is described in the following.

Definition 3.4.6 Generalized Rule Graph. *Let T be a transactional Twitter dataset and θ be taxonomy built over term in T. Let minsup and minconf be minimum thresholds. Let R be the set of generalized rules satisfying both minsup and minconf. A generalized rule graph G is a graph whose nodes represents frequent generalized itemsets being the antecedent and consequence of at least one rule in R. For each generalized rule $A \Rightarrow B$ contained in R, there exists an edge in G from A to B.*

The analyzed dataset being a sparse and high dimensional produces a large set of potentially frequent generalized rules. To ease the analysts, TGRV tool provides three complementary views of the *Generalized Rule Graph*, which may help in navigation and visualization of the mined results.

Item-constrained viewpoint This viewpoint allows to pay attention on the subsets of the rules containing specific item combinations. For instance, consider a visualization constraint such as $\{New\ York\ State\} \Rightarrow \{*\}$. It states all rules containing *New York State* as antecedent and every other item combination as consequent should be visualized. Hence, recalling the previous example of tweet transaction, a rule like $\{New\ York\ State\} \Rightarrow \{transportation\}$ is provided as the result.

Level-constrained viewpoint The analysts are allowed to explore in this viewpoint all the rule having items with common generalization level.

For example, consider the taxonomy depicted in Fig. 3.6. Visualizing only the rules that are composed of at least one item of level 3 may allow selecting rules like $\{USA\} \Rightarrow \{transportation\}$. By considering rules having only items of level 2, instead may result in rule like $\{France\} \Rightarrow \{(9\ a.m.\ to\ 12\ a.m.\}$.

Item- and level-constrained viewpoint This viewpoint is the combination of previous viewpoints that allows to visualize the specific terms and at specific abstraction levels. For instance, consider again $\{New\ York\ State\} \Rightarrow \{*\}$, the expert may be willing to explore rules containing a specific generalized term or any of its descendants. Thus, rules including $\{New\ York\ City\} \Rightarrow \{transportation\}$ are held relevant and should be visualized.

The constraints selection to visualize the generalized rules depends on the characteristics of the analyzed textual data and their abstraction levels available. The provision of easily navigation form one viewpoint to another of the visualization model allows analysts to deeply understand and analyze the correlations of targeted social network data at number of different abstraction levels. Hence, experts are fascinated for exploring data correlations at higher generalized concepts and possibly deepen down the exploration till interested term correlations are detected, even when dealing with sparse textual data.

3.4.2 Experimental results

The effectiveness of the TRGV approach is assessed in this section, which describes a set of experiments performed on two real-world datasets published on social network website (i.e., Twitter). The detailed evaluation is reported in the following.

3.4.2.1 Evaluated datasets and taxonomy

The crawler accessed Twitter's publicly available stream of tweet data over a time period [7th September 2012, 23rd September 2012]. The tweets were selected according to different topics (e.g., education, social events, etc). The collected tweets data is then processed (see Section 3.4.1.1).

The TRGV injected the tweet terms with their corresponding higher abstraction levels upto level-5, in order to keep the dataset easily manageable, otherwise the abstraction levels may be achieved depending on the terms and their available generalized set of terms in WordNet [169]. Two different transactional Twitter datasets has been generated from the crawler data

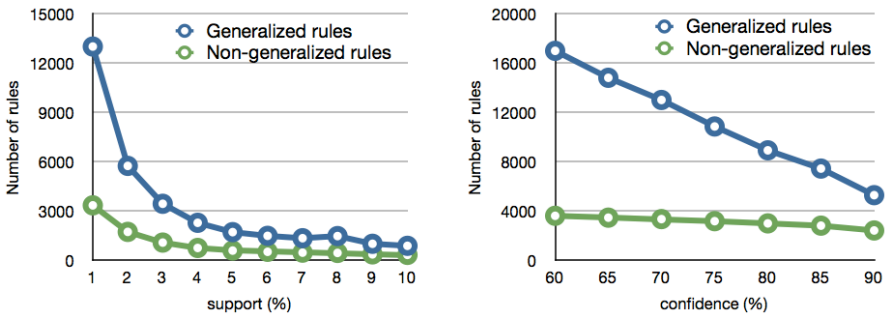
Table 3.6: The characteristics of transactional Twitter dataset

Name of dataset	*No. of transactions*	*Distinct terms*	*Min. terms per transaction*	*Avg. terms per transaction*	*Max. terms per transaction*
Social	634	3263	1	15	109
Politics	140	577	1	14	79

stream namely *Social* and *Politics*, which cover different topics such as social theme and political events. The characteristics of the considered datasets are reported in Table 3.6.

3.4.2.2 The characteristics of the mined rules

The proposed system architecture (i.e., TGRV) allows to visualize frequent generalized rules extracted from the considered transactional Twitter datasets. The generation of the generalized rules are directly affected from the enforcement of the minimum support and confidence thresholds. The impact of the mined rules with respect to both mining constraints minimum support and confidence are analyzed and reported in Fig. 3.7 for the considered dataset, namely *Politics*.



(a) Impact of the support threshold (b) Impact of the confidence threshold

Figure 3.7: Politics dataset. Impact of support and confidence on the number of mined rules.

The number of mined rules form the real-world dataset (i.e., *Politics*), taken as representative, are reported in Fig. 3.7(a) by varying the minimum support threshold in the range [1%, 10%], whilst minimum confidence is set to 70% (i.e., *minconf=70%*). Likewise, Fig. 3.7(b) depicts the variation in minimum confidence in the range [60%, 90%], when minimum support threshold equals to 3% (i.e., *minsup=3%*). The number of generalized rules are significantly increased as expected by lowering the *minsup* due to combinational

increase in the set of terms and their abstraction levels, where as the *minconf* seems to be more limited during the medium support threshold values. Fig. 3.7 represents the extracted rules in generalized and non-generalized ones (i.e., rules containing at least one generalized term and no any generalized term). The cardinality of the set of mined rules for both generalized and non-generalized is comparable at relatively high support threshold values (i.e., *minsup=8%*). Instead at lower the *minsup*, a significant number of non-generalized rules become infrequent, where as generalized rules are still mined due to the fact that generalized rules contain more general information. Thus, the gap between the number of both mined rules (i.e., generalized and non-generalized) increases significantly (e.g., about 80% increase in generalized rules in comparison of 20% non-generalized rules at *minsup=1%*).

3.4.2.3 A real-life use-case study

A real-life use-case as been reported in this section for the effectiveness of the TRGV system targeting to a topic trend analysis. The validation of the discovered knowledge and its usefulness has been carried out with the help of domain expert.

Consider, for instance an application scenario for the TRGV system, a domain expert in-charge of analyzing Twitter posts for the discovery of topical interest. The rules like reported in Fig. 3.8 may suggest experts that education is currently the matter of great concern on the web. Hence, it leads towards the fact that Twitter users are likely to be more interested in news, articles and blogs related to schooling and teaching. It is worth mentioning that non-generalized rules such as $\{Student\} \Rightarrow \{School\}$ and $\{Person\} \Rightarrow \{Educational\ institution\}$ are detected at relatively lower mining parameters (i.e., *minsup=3%* and *minconf=70%*). Whilst, lowering down the *minsup=1%* more specific rules may become infrequent, and thus, are discarded. However, on the contrary generalized rules are still kept in the mined results. Hence, the information related to more general topic is still maintained.

Visualizing rules related to specific political events (e.g., American Presidential elections, European Union meetings) by considering *Politics* dataset may allow experts to better understand the interest of the online community (i.e., Web users) about specific topics. For example, a number of rules containing keywords `President` and `Vote` are extracted leading towards the fact that online users are focused on the U.S.A. Presidential elections. Although the analyzed collection of data is sparse, the TGRV allows experts

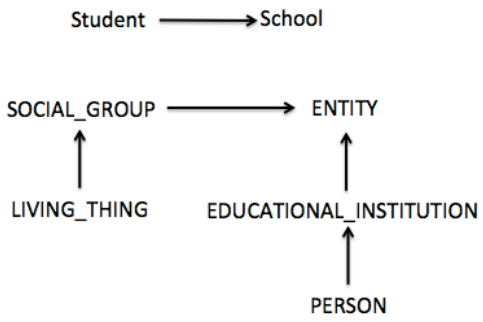


Figure 3.8: Social dataset. Portion of the Generalized Rule Graph. minsup=3% minconf=70%

to explore high level term correlations holding valuable information for targeted actions. Lets consider even if the correlation between the keywords `President` and `New York City` is infrequent in the analyzed dataset enforcing lower support threshold (i.e., *minsup=1%*), the corresponding high level correlation in between `President` and `USA` is deemed extracted. Therefore, visualization of generalized association rules is effective and useful instead of conventional approaches of rule mining, particularly, when data distribution of the considered dataset is sparse.

Chapter 4

Conclusion and future works

The PhD research work focused on the analysis of data coming from complex application domains using data mining techniques. In particular, the following real data relative to two complex application domains namely healthcare data and textual data are analysed. The experimental results on the analysed data highlight the effectiveness of the proposed data mining approaches.

Health is one of the essential elements in human-lives. Thus, correct medical treatments need to be carefully adopted to cure diseases properly and accurately on timely basis. Practitioners and medical experts require error-free and in-time information to prescribe the potentially correct treatments. However, healthcare problems are complex, diverse, and dispersed. It is complicated matter to manually investigate the medical history of each individual, since each person has his/her own medical history and disease conditions. Furthermore, the main difficulty evaluating the effectiveness of the diagnostic systems and adequacy of diagnostic procedure is tracing the path (i.e., patterns) followed by patients.

The proposed approaches in the PhD activity for healthcare data analysis address the identification of the medical pathways followed by patients for different pathologies. In addition, they also evaluate the applicability of diagnostic guidelines in healthcare systems and demonstrate how to extract frequent diagnostic pathways followed by patients for the improvement of existing guidelines. The achieved results reflect the actual treatment procedures and show the diversity of the adopted medical pathways reflecting the seriousness of the disease for patients. The adopted approaches not only highlight the interdependency between various diagnostic examinations but also identify subgroups of the patients with similar medical pathways for a given pathology. The discovered information may help for both improving

treatment procedures and managing healthcare resources to provide better care. Future research activity in this field will investigate the medical pathways under certain timing constraints to address the frequent pathways for a given time period in a given pathology. For example, given a month time period, the treatment procedures followed by patients with a given dataset. Discovering subgroups of patients with different behaviours allows prescribing better care treatments and organising healthcare resources efficiently.

The research activities in textual data analysis followed two parallel directions: (i) proposing a novel summarization approach to encapsulate large document collections and (ii) exploiting generalization rules to highlight significant high level data correlations.

In the context of multi-document summarization, a general-purpose and novel graph-based summarizer (GraphSum) has been developed. The GraphSum discovers and combines frequent itemsets to represent correlations among multiple terms neglected by previous approaches in a correlation graph. The ranking of the sentences, which are selected to include in summary adopts graph ranking strategy that distinguish between positive and negative correlations. The effectiveness of the GraphSum summarizer has been reflected in experimental results on real-world data as well as on benchmark document collections. Experiments verified that the GraphSum better performed than that of the state-of-the-art approaches and it also significantly outperformed the approaches that rely on semantics-based analysis, though it does not require semantics-based analysis. The ability of the summarizers to adopt the generated summaries in accordance with actual user interests, and to update them dynamically, when documents are added/removed in the initial set of documents is one of the challenging issue in multi-document summarization. Hence, as a future work, investigation of the use of user-generated contents published on social networks and online communities will be considered to derive the summarization process, and to analyse the evolution of the extracted patterns over time as an application of dynamic itemset miners.

In the context of data visualization, a tool namely Twitter Generalized Rules Visualizer (TRGV) has been developed to address the generalized association rule mining from textual data contents published on social network website (i.e., user-generated contents or tweets) and their visualization. The TRGV is a graph-based visualization model, which allows exploring the correlations among tweet terms as well as their higher abstraction levels. It offers multi-faceted viewpoints to visualize and investigate in-depth the target social network data. The performance of the TGRV has been evaluated on the real transactional twitter datasets. The usefulness and effectiveness

of the tool is investigated in different use-cases such as topic trend detection. As future work, the proposed approach is intended to be applied in diverse application domains composed of sparse data such as healthcare and sports data.

List of Figures

List of Tables

Bibliography

[1] E. AbuKhousa and P. Campbell. Predictive data mining to support clinical decisions an overview of heart disease prediction systems. In *Innovations in Information Technology (IIT), 2012 International Conference on*, pages 267 –272, march 2012.

[2] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. Mining association rules between sets of items in large databases. *SIGMOD Rec.*, 22(2):207–216, June 1993.

[3] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules in large databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, pages 487–499, San Francisco, CA, USA, 1994. Morgan Kaufmann Publishers Inc.

[4] Rakesh Agrawal and Ramakrishnan Srikant. Mining sequential patterns. In *Proceedings of the Eleventh International Conference on Data Engineering*, ICDE '95, pages 3–14, Washington, DC, USA, 1995. IEEE Computer Society.

[5] D.A. Alexandrou, I.E. Skitsas, and G.N. Mentzas. A holistic environment for the design and execution of self-adaptive clinical pathways. In *Information Technology and Applications in Biomedicine, 2009. ITAB 2009. 9th International Conference on*, pages 1 –5, nov. 2009.

[6] M.K. Ali, S. Shah, and N. Tandon. Review of electronic decision-support tools for diabetes care: a viable option for low- and middle-income countries? *J Diabetes Sci Technol*, 5(3):553–70, 2011.

[7] ASCRS. Practice parameters for the treatment of rectal carcinoma, 1999. *http://www.fascrs.org/files/rectal_cancer_0605.pdf*
Last accessed: 24, September 2012.

[8] Hrvoje Bacan, Igor S. P, and Darko Gulija. Automated news item categorization. In *Proceedings of the 19th Annual Conference of The Japanese Society for Artificial Intelligence*, pages 251–256. Springer-Verlag, 2005.

[9] E. Baralis, L. Cagliero, T. Cerquitelli, V. D'Elia, and P. Garza. Support driven opportunistic aggregation for generalized itemset extraction. In *Intelligent Systems (IS), 2010 5th IEEE International Conference*, pages 102 –107, july 2010.

[10] Elena Baralis, Giulia Bruno, Silvia Chiusano, Virna C. Domenici, Naeem A. Mahoto, and Caterina Petrigni. Analysis of medical pathways by means of frequent closed sequences. In *Proceedings of the 14th international conference on Knowledge-based and intelligent information and engineering systems: Part III*, KES'10, pages 418–425, Berlin, Heidelberg, 2010. Springer-Verlag.

[11] Elena Baralis, Luca Cagliero, Tania Cerquitelli, and Paolo Garza. Generalized association rule mining with constraints. *Inf. Sci.*, 194:68–84, July 2012.

[12] Elena Baralis, Luca Cagliero, Saima Jabeen, and Alessandro Fiori. Multi-document summarization exploiting frequent itemsets. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, SAC '12, pages 782–786, New York, NY, USA, 2012. ACM.

[13] Regina Barzilay and Michael Elhadad. Using lexical chains for text summarization. In *In Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, pages 10–17, 1997.

[14] Hila Becker, Mor Naaman, and Luis Gravano. Selecting quality twitter content for events. In Lada A. Adamic, Ricardo A. Baeza-Yates, and Scott Counts, editors, *ICWSM'11*. The AAAI Press, 2011.

[15] Riccardo Bellazzi and Blaz Zupan. Predictive data mining in clinical medicine: Current issues and guidelines. *International journal of medical informatics*, 77(2):81–97, Feb 2008.

[16] Sabine Bergler, RenÃ© Witte, Michelle Khalife, Zhuoyan Li, and Frank Rudzicz. Using knowledge-poor coreference resolution for text summarization. In *in DUC, Workshop on Text Summarization, May-June*, pages 85–92, 2003.

[17] Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly, Beijing, 2009.

[18] Julien Blanchard, Fabrice Guillet, and Henri Briand. Exploratory visualization for association rule rummaging. In *KDD-03 Workshop on Multimedia Data Mining (MDM-03*, pages 107–114, 2003.

[19] Johan Bollen, Huina Mao, and Alberto Pepe. Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In

In proceddings Fifth International AAAI Conference on Weblogs and Social Media, pages 450–453, Barcelona, Spain, 2011.

[20] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

[21] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Comput. Netw. ISDN Syst.*, 30(1-7):107–117, April 1998.

[22] A.L. Buczak, L.J. Moniz, B.H. Feighner, and J.S. Lombardo. Mining electronic medical records for patient care patterns. In *Computational Intelligence and Data Mining, 2009. CIDM '09. IEEE Symposium on*, pages 146 –153, april 2009.

[23] Keith D. Calligaro, Matthew J. Dougherty, Carol A. Raviola, David J. Musser, and Dominic A. DeLaurentis. Impact of clinical pathways on hospital costs and early outcome after major vascular surgery. *Journal of Vascular Surgery*, 22(6):649 – 660, 1995.

[24] Effective Health Care. The management of colorectal cancer, 1997. *http://www.york.ac.uk/inst/crd/EHC/ehc36.pdf* Last accessed: 24, September 2012.

[25] Effective Health Care. Getting evidence into practice, 1999. *http://www.york.ac.uk/inst/crd/EHC/ehc51.pdf* Last accessed: 24, September 2012.

[26] Patricia B. Cerrito. Mining the electronic medical record to examine physician decisions. In *Advanced Computational Intelligence Paradigms in Healthcare*, pages 113–126. 2007.

[27] Deepayan Chakrabarti and Kunal Punera. Event summarization using tweets. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM)*, july 2011.

[28] Sharma Chakravarthy and Hongen Zhang. Visualization of association rules over relational dbmss. In *Proceedings of the 2003 ACM symposium on Applied computing*, SAC '03, pages 922–926, New York, NY, USA, 2003. ACM.

[29] Ramnath Chellappa. Intermediaries in Cloud-Computing: A New Computing Paradigm. *INFORMS*, 1997.

[30] Chiehfeng Chen, Kung Chen, Chien-Yeh Hsu, Wen-Ta Chiu, and Yu-Chuan (Jack) Li. A guideline-based decision support for pharmacological treatment can improve the quality of hyperlipidemia management. *Comput. Methods Prog. Biomed.*, 97(3):280–285, March 2010.

[31] Yu Chen, Lars Henning Pedersen, Wesley W. Chu, and Jorn Olsen. Drug exposure side effects from mining pregnancy data. *SIGKDD Explor. Newsl.*, 9(1):22–29, June 2007.

[32] Marc Cheong and Vincent Lee. Integrating web-based intelligence retrieval and decision-making from the twitter trends knowledge base. In *Proceedings of the 2nd ACM workshop on Social web search and mining*, SWSM '09, pages 1–8, New York, NY, USA, 2009. ACM.

[33] Wesley W. Chu. Challenges and techniques for mining real clinical data. In *GrC*, pages 2–4, 2008.

[34] Wesley T. Chuang and Jihoon Yang. Extracting sentence segments for text summarization: a machine learning approach. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '00, pages 152–159, New York, NY, USA, 2000. ACM.

[35] John M. Conroy, Jade Goldstein, Judith D. Schlesinger, and Dianne P. O'leary. Left-brain/right-brain multi-document summarization. In *In Proceedings of the Document Understanding Conference (DUC*, 2004.

[36] John M. Conroy, Judith D. Schlesinger, Jeff Kubina, Peter A. Rankel, and Dianne P. O'Leary. Classy 2011 at tac: Guided and multi-lingual summaries and evaluation metrics. In *Proceedings of Text Analysis Conference (TAC 2011)*, 2011.

[37] Olivier Couturier, Tarek Hamrouni, Sadok Ben Yahia, and Engelbert Mephu Nguifo. A scalable association rule visualization towards displaying large amounts of knowledge. In *Proceedings of the 11th International Conference Information Visualization*, IV '07, pages 657–663, Washington, DC, USA, 2007. IEEE Computer Society.

[38] Dipanjan Das and Andre' F. T. Martins. A survey on automatic text summarization, nov 2007.

[39] DBSCAN. Dbscan pseudocode, 2013. Online: *http://en.wikipedia.org/wiki/DBSCAN*
Accessed on 25 February 2013.

[40] Alberto Diaz and Pablo Gervas. User-model based personalized summarization. *Inf. Process. Manage.*, 43(6):1715–1734, November 2007.

[41] Thomas G. Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Comput.*, 10(7):1895–1923, October 1998.

[42] Marcos AurÃ©lio Domingues and Solange Oliveira Rezende. Using taxonomies to facilitate the analysis of the association rules. *CoRR*, abs/1112.1734, 2011.

[43] Sumeet Dua, Michael P. Dessauer, and Prerna Sethi. Evaluating cluster preservation in frequent itemset integration for distributed databases. *J. Med. Syst.*, 35(5):845–853, October 2011.

[44] Document Understanding Conference (DUC), 2004.
Online: *http://www-nlpir.nist.gov/projects/duc/pubs.html*.

[45] Margaret H. Dunham. *Data Mining: Introductory and Advanced Topics*. Prentice Hall, 1 edition, September 2002.

[46] Christoph F. Eick, Nidal Zeidat, and Zhenghong Zhao. Supervised clustering algorithms and benefits. In *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence*, ICTAI '04, pages 774–776, Washington, DC, USA, 2004. IEEE Computer Society.

[47] Haytham Elghazel, Veronique Deslandres, Mohand-Said Hacid, Alain Dussauchoy, and Hamamache Kheddouci. A new clustering approach for symbolic data and its validation: Application to the healthcare data. In Floriana Esposito, Zbigniew Ras, Donato Malerba, and Giovanni Semeraro, editors, *Foundations of Intelligent Systems*, volume 4203 of *Lecture Notes in Computer Science*, pages 473–482. Springer Berlin / Heidelberg, 2006.

[48] Gunes Erkan and Dragomir R. Radev. Lexrank: graph-based lexical centrality as salience in text summarization. *J. Artif. Int. Res.*, 22(1):457–479, December 2004.

[49] Thomas Erl. *Service-Oriented Architecture: Concepts, Technology, and Design*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2005.

[50] Martin Ester, Hans peter Kriegel, Jörg S, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. pages 226–231. AAAI Press, 1996.

[51] Brian Everitt. *Cluster Analysis*. Halsted Press, London; New York, second edition, 1980.

[52] Usama Fayyad, Georges G. Grinstein, and Andreas Wierse. *Information Visualization in Data Mining and Knowledge Discovery*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st edition, 2001.

[53] Usama M. Fayyad. Data mining and knowledge discovery: Making sense out of data. *IEEE Expert: Intelligent Systems and Their Applications*, 11(5):20–25, October 1996.

[54] Usama M. Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. Advances in knowledge discovery and data mining. chapter From data mining to knowledge discovery: an overview, pages 1–34. American Association for Artificial Intelligence, Menlo Park, CA, USA, 1996.

[55] Oi Mean Foong, Alan Oxley, and Suziah Sulaiman. Challenges and trends of automatic text summarization. *International Journal of Information and Telecommunication Technology*, 1(1):34–39, 2010.

[56] Institute for Clinical System Improvement. Health care guideline: Colorectal cancer screening, 2010. *http://www.icsi.org/colorectal_cancer_screening/colorectal_cancer_screening_5.html* Last accessed: 24, September 2012.

[57] National Institute for Health and Clinical Excellence. Antenatal care: routine care for the healthy pregnant woman - nice cg6. Technical report, 2003.

[58] Kavita Ganesan, ChengXiang Zhai, and Jiawei Han. Opinosis a graph based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 340–348, 2010.

[59] Fabien Girardin, Francesco Calabrese, Filippo Dal Fiore, Carlo Ratti, and Josep Blat. Digital footprinting: Uncovering tourists with user-generated content. *IEEE Pervasive Computing*, 7(4):36–43, oct 2008.

[60] Jade Goldstein, Vibhu Mittal, Jaime Carbonell, and Mark Kantrowitz. Multi-document summarization by sentence extraction. In *Proceedings of the 2000 NAACL-ANLPWorkshop on Automatic summarization - Volume 4*, NAACL-ANLP-AutoSum '00, pages 40–48, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

[61] Yihong Gong. Generic text summarization using relevance measure and latent semantic analysis. In *in Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New Orleans, Louisiana, USA, 2001.

[62] Anjana. Gosain and Amit. Kumar. Analysis of health care data using different data mining techniques. In *Intelligent Agent Multi-Agent Systems, 2009. IAMA 2009. International Conference on*, pages 1–6, july 2009.

[63] Nizar Grira, Michel Crucianu, and Nozha Boujemaa. Unsupervised and semi-supervised clustering: a brief survey. In *in A Review of Machine Learning Techniques for Processing Multimedia Content, Report of the MUSCLE European Network of Excellence (FP6*, 2004.

[64] Vishal Gupta and Gurpreet Singh Lehal. A survey of text summarization extractive techniques. *Journal of Emerging Technologies in Web Intelligence*, 2(3):258–268, 2010.

[65] Jiawei Han and Yongjian Fu. Mining multiple-level association rules in large databases. *IEEE Trans. on Knowl. and Data Eng.*, 11(5):798–805, September 1999.

[66] Jiawei Han, Micheline Kamber, and Jian Pei. *Data Mining: Concepts and Techniques, Second Edition (The Morgan Kaufmann Series in Data Management Systems)*. Morgan Kaufmann, San Francisco, CA, USA, 2 edition, January 2006.

[67] J. Michael Hardin and David C. Chhieng. Data mining and clinical decision support systems, 2007.

[68] Susan Havre, Elizabeth Hetzler, Paul Whitney, and Lucy Nowell. Themeriver: Visualizing thematic changes in large document collections. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):9–20, January 2002.

[69] Jochen Hipp, Andreas Myka, Rüdiger Wirth, and Ulrich Güntzer. A new algorithm for faster mining of generalized association rules. In *Proceedings of the Second European Symposium on Principles of Data Mining and Knowledge Discovery*, PKDD '98, pages 74–82, London, UK, UK, 1998. Springer-Verlag.

[70] Tu Anh Nguyen Hoang, Hoang Khai Nguyen, and Quang Vinh Tran. An efficient vietnamese text summarization approach based on graph

model. In *Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2010 IEEE RIVF International Conference on*, pages 1–6, nov. 2010.

[71] C. Honey and S.C. Herring. Beyond microblogging: Conversation and collaboration via twitter. In *System Sciences, 2009. HICSS '09. 42nd Hawaii International Conference on*, pages 1 –10, jan. 2009.

[72] David W. Hosmer and Stanley Lemeshow. *Applied logistic regression (Wiley Series in probability and statistics) 2nd ed.* Wiley, 2000.

[73] Zhengxing Huang, Xudong Lu, and Huilong Duan. Using recommendation to support adaptive clinical pathways. *J. Med. Syst.*, 36(3):1849–1860, June 2012.

[74] Kian huat Ong, Kok leong Ong, Wee-Keong Ng, and Ee-Peng Lim. Crystalclear: Active visualization of association rules. In *In ICDM'02 International Workshop on Active Mining AM2002*. Press, 2002.

[75] IBM. Intelligent information systems (aka quest), 2012.
online: *http://http://www.almaden.ibm.com/cs/disciplines/iis/*.

[76] David Isern and Antonio Moreno. Computer-based execution of clinical guidelines: A review. *International Journal of Medical Informatics*, 77(12):787–808, 2008.

[77] Mark W. Isken and Balaji Rajagopalan. Data mining to support simulation modeling of patient flow in hospitals. *Journal of Medical Systems*, 26:179–197, 2002. 10.1023/A:1014814111524.

[78] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31(3):264–323, September 1999.

[79] Bernard J. Jansen, Mimi Zhang, Kate Sobel, and Abdur Chowdury. Twitter power: Tweets as electronic word of mouth. *J. Am. Soc. Inf. Sci. Technol.*, 60(11):2169–2188, nov 2009.

[80] Maher Jaoua and Abdelmajid Ben Hamadou. Automatic text summarization of scientific articles based on classification of extract's population. In *Proceedings of the 4th international conference on Computational linguistics and intelligent text processing*, CICLing'03, pages 623–634, Berlin, Heidelberg, 2003. Springer-Verlag.

[81] Stephen Johnson. Hierarchical clustering schemes. *Psychometrika*, 32:241–254, 1967. 10.1007/BF02289588.

[82] B.-H. Juang and L.R. Rabiner. The segmental k-means algorithm for estimating parameters of hidden markov models. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 38(9):1639 –1641, sep 1990.

[83] L. Kaufman and P. J. Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. John Wiley and Sons, New York, 1990.

[84] Reza S. Kazemzadeh and Kamran Sartipi. Incoporating data mining applications into clinical guildelines. In *Computer-Based Medical Systems, 2006. CBMS 2006. 19th IEEE International Symposium on*, pages 321 –328, 0-0 2006.

[85] Daniel A. Keim. Designing pixel-oriented visualization techniques: Theory and applications. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):59–78, January 2000.

[86] Daniel A. Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1–8, January 2002.

[87] Benjamin King. Step-wise clustering procedures. *Journal of the American Statistical Association*, 62(317):86–101, 1967.

[88] Carson Kai-Sang Leung, Pourang P. Irani, and Christopher L. Carmichael. Wifisviz: Effective visualization of frequent itemsets. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pages 875–880, Washington, DC, USA, 2008. IEEE Computer Society.

[89] Xin Li, Lei Guo, and Yihong Eric Zhao. Tag-based social interest discovery. In *Proceedings of the 17th international conference on World Wide Web*, WWW '08, pages 675–684, New York, NY, USA, 2008. ACM.

[90] Zhenhui Li, Jiawei Han, Ming Ji, Lu-An Tang, Yintao Yu, Bolin Ding, Jae-Gil Lee, and Roland Kays. Movemine: Mining moving object data for discovery of animal movement patterns. *ACM Trans. Intell. Syst. Technol.*, 2(4):37:1–37:32, July 2011.

[91] Chin-Yew Lin and Eduard Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*,

NAACL '03, pages 71–78, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[92] Fu-ren Lin, Shien-chao Chou, Shung-mei Pan, and Yaomei Chen. Mining time dependency patterns in clinical pathways. *International Journal of Medical Informatics*, 62(1):11–25, June 2001.

[93] Fu-ren Lin and Chia-Hao Liang. Storyline-based summarization for news topic retrospection. *Decis. Support Syst.*, 45(3):473–490, June 2008.

[94] Marina Litvak and Mark Last. Graph-based keyword extraction for single-document summarization. In *Proceedings of the Workshop on Multi-source Multilingual Information Extraction and Summarization*, MMIES '08, pages 17–24, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[95] Yan Liu and Gavriel Salvendy. Visualization to facilitate association rules modelling: A review. *Ergonomia IJE&HF*, 27(1):11–23, 2005.

[96] Elena Lloret and Manuel Palomar. Challenging issues of automatic summarization: relevance detection and quality-based evaluation. *Slovenian Society Informatika*, 34(1):29–35, 2010.

[97] H. P. Luhn. The automatic creation of literature abstracts. *IBM J. Res. Dev.*, 2(2):159–165, April 1958.

[98] Inderjeet Mani, David House, Gary Klein, Lynette Hirschman, Therese Firmin, and Beth Sundheim. The tipster summac text summarization evaluation. In *Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics*, EACL '99, pages 77–85, Stroudsburg, PA, USA, 1999. Association for Computational Linguistics.

[99] Benjamin M. Marlin, David C. Kale, Robinder G. Khemani, and Randall C. Wetzel. Unsupervised pattern discovery in electronic health care data using probabilistic clustering models. In *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*, IHI '12, pages 389–398, New York, NY, USA, 2012. ACM.

[100] Laura Maruşter, Ton Weijters, Geerhard de Vries, Antal van den Bosch, and Walter Daelemans. Logistic-based patient grouping for multi-disciplinary treatment. *Artif. Intell. Med.*, 26(1-2):87–107, September 2002.

[101] Michael Mathioudakis and Nick Koudas. Twittermonitor: trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, SIGMOD '10, pages 1155–1158, New York, NY, USA, 2010. ACM.

[102] Geoffrey J. McLachlan and T. Krishnan. *The EM algorithm and extensions / Geoffrey J. McLachlan, Thriyambakam Krishnan.* Wiley, New York :, 1997.

[103] Ho Si Meng and Simon Fong. Visualizing e-government portal and its performance in webvs. In *Fifth International Conference on Digital Information Management ICDIM'10*, pages 315–320, 2010.

[104] Yajie Miao and Chunping Li. Wikisummarizer - a wikipedia-based summarization system. In *Proceedings of Text Analysis Conference (TAC 2010)*, 2010.

[105] Ingo Mierswa, Michael Wurst, Ralf Klinkenberg, Martin Scholz, and Timm Euler. Yale: Rapid prototyping for complex data mining tasks. In Lyle Ungar, Mark Craven, Dimitrios Gunopulos, and Tina Eliassi-Rad, editors, *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 935–940, New York, NY, USA, August 2006. ACM.

[106] Rada Mihalcea and Paul Tarau. A language independent algorithm for single and multiple document summarization. In *In Proceedings of IJCNLP'2005*, Korea, 2005.

[107] Mandar Mitra, Amit Singhal, and Chris Buckley. Automatic text summarization by paragraph extraction. In *Workshop On Intelligent Scalable Text Summarization*, Madrid, Spain, 1997.

[108] Naresh Kumar Nagwani and Shrish Verma. A frequent term and semantic similarity based single document text summarization algorithm. *International Journal of Computer Applications*, 17(2):36–40, March 2011. Published by Foundation of Computer Science.

[109] NCI. Colon cancer-(pdq)-treatment-health professionals, guidelines, 2001. *http://www.cancernet.nci.nih.gov/pdq/pdq_treatment.shtml* Last accessed: 28, September 2010.

[110] New York State Department of Health (DOH). New york state medicaid update. *The official newsletter of the New York Medicaid Program. Prenatal Care Special Edition*, 26(2), 2010.

[111] Royal College of Surgeon of England. Guidelines for the management of colorectal cancer, 1996.
http://www.acpgbi.org.uk Last accessed: 28, September 2010.

[112] Manabu Okumura, Takahiro Fukusima, Hidetsugu Nanba, and Tsutomu Hirao. Text summarization challenge 2 text summarization evaluation at ntcir workshop 3. *SIGIR Forum*, 38(1):29–38, July 2004.

[113] Free online searchable. Icd-9-cm, 2009.
http://icd9cm.chrisendres.com Last accessed: 24, September 2012.

[114] Cancer Care Ontario. Colonoscopy standards, 2007.
https://www.cancercare.on.ca/common/pages/UserFile.aspx?fileId=33457
Last accessed: 24, September 2012.

[115] C. D. Paice. Constructing literature abstracts by computer: techniques and prospects. *Inf. Process. Manage.*, 26(1):171–186, apr 1990.

[116] Sellappan Palaniappan and Chua Sook Ling. Clinical decision support using olap with data mining. *IJCSNS International Journal of Computer Science and Network Security*, 8(9):290–296, September 2008.

[117] M. Panella, S. Marchisio, and F. Di Stanislao. Reducing clinical variations with clinical pathways: do pathways work? *International Journal for Quality in Health Care*, 15(6):509–522, 2003.

[118] Nicolas Pasquier, Yves Bastide, Rafik Taouil, and Lotfi Lakhal. Discovering frequent closed itemsets for association rules. In *Proceedings of the 7th International Conference on Database Theory*, ICDT '99, pages 398–416, London, UK, UK, 1999. Springer-Verlag.

[119] Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, Helen Pinto, Qiming Chen, Umeshwar Dayal, and Mei-Chun Hsu. Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In *Proceedings of the 17th International Conference on Data Engineering*, ICDE '01, pages 215–, Washington, DC, USA, 2001. IEEE Computer Society.

[120] J. Ross Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[121] Dragomir R Radev, Hongyan Jing, Malgorzata Stys, and Daniel Tam. Centroid-based summarization of multiple documents. *Inf. Process. Manage.*, 40(6):919–938, nov 2004.

[122] T. Ralphs and M. Guzelsoy. The symphony callable library for mixed integer programming. *The Next Wave in Computing, Optimization, and Decision Technologies*, 29:61–76, 2006.
Software available at *http://www.coin-or.org/SYMPHONY*.

[123] Krishnan Ramanathan, Yogesh Sankarasubramaniam, Nidhi Mathur, and Ajay Gupta 0005. Document summarization using wikipedia. In Uma Shanker Tiwary, Tanveer J. Siddiqui, M. Radhakrishna, and M. D. Tiwari, editors, *IHCI*, pages 254–260. Springer India, 2009.

[124] W.M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

[125] Commissione Oncologica Regionale. Tumori del colon retto. linee guida cliniche organizzative della regione piemonte., 2001.
http://www.cpo.it Last accessed: 24, September 2012.

[126] J. Roberto and J.r. Bayardo. Efficiently mining long patterns from databases. In Laura M. Haas and Ashutosh Tiwary, editors, *SIGMOD 1998*, pages 85–93, 1998.

[127] Lior Rokach and Oded Maimon. Clustering methods. In Oded Maimon and Lior Rokach, editors, *The Data Mining and Knowledge Discovery Handbook*, pages 321–352. Springer, 2005.

[128] H.C. Romesburg. *Cluster Analysis for Researchers*. Lulu Press, Morrisville, North Carolina, 2004. Reprint of 1984 edition, with minor revisions.

[129] Nadav Rotem. Open text summarizer (ots), 2003.
Software available at *http://libots.sourceforge.net*.

[130] Peter Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.*, 20(1):53–65, November 1987.

[131] Gerard Salton, Amit Singhal, Mandar Mitra, and Chris Buckley. Automatic text structuring and summarization. *Inf. Process. Manage.*, 33(2):193–207, March 1997.

[132] Jochen Schuld, Thilo SchÃ¤fer, Stefan Nickel, Peter Jacob, Martin K. Schilling, and Sven Richter. Impact of it-supported clinical pathways on medical staff satisfaction. a prospective longitudinal cohort study. *I. J. Medical Informatics*, pages 151–156, 2011.

[133] sgi. Silicon graphics international corp, 2012.
online: *http://www.sgi.com/products/software/?/mineset*.

[134] R. Sharan, A. Maron-Katz, and R. Shamir. Click and expander: a system for clustering and visualizing gene expression data. *Bioinformatics*, 19(14):1787–1799, 2003.

[135] Beaux Sharifi, Mark-Anthony Hutton, and Jugal Kalita. Automatic summarization of twitter topics. In *National Workshop on Design and Analysis of Algorithms, NWDAA'10*, Tezpur University, Assam, India, 2010.

[136] Chia-Ping Shen, Chinburen Jigjidsuren, Sarangerel Dorjgochoo, Chi-Huang Chen, Wei-Hsin Chen, Chih-Kuo Hsu, Jin-Ming Wu, Chih-Wen Hsueh, Mei-Shu Lai, Ching-Ting Tan, Erdenebaatar Altangerel, and Feipei Lai. A data-mining framework for transnational healthcare system. *Journal of Medical Systems*, 36:2565–2575, 2012.

[137] M. Shouman, T. Turner, and R. Stocker. Using data mining techniques in heart disease diagnosis and treatment. In *Electronics, Communications and Computers (JEC-ECC), 2012 Japan-Egypt Conference on*, pages 173 –177, march 2012.

[138] Jawed Siddiqi, Babak Akhgar, Alicja Gruzdz, Ghasem Zaefarian, and Aleksandra Ihnatowicz. Automated diagnosis system to support colon cancer treatment: Match. In *Proceedings of the Fifth International Conference on Information Technology: New Generations*, ITNG '08, pages 201–205, Washington, DC, USA, 2008. IEEE Computer Society.

[139] Evangelos Simoudis. Reality check for data mining. *IEEE Expert*, 11(5):26–33, 1996.

[140] P. H. A. Sneath. Some thoughts on bacterial classification. *J Gen Microbiol*, 17(1):184–200, aug 1957.

[141] P.H.A. Sneath and R.R. Sokal. *Numerical Taxonomy. The Principles and Practice of Numerical Classification*. Freeman, 1973.

[142] SPSS. The spss twostep cluster component, 2001.
Technical report, online: *http://www.spss.ch/*.

[143] Ramakrishnan Srikant and Rakesh Agrawal. Mining generalized association rules. In *Proceedings of the 21th International Conference on Very Large Data Bases*, VLDB '95, pages 407–419, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[144] K. Srinivas, G.R. Rao, and A. Govardhan. Analysis of coronary heart disease and prediction of heart attack in coal mining regions using data mining techniques. In *Computer Science and Education (ICCSE), 2010 5th International Conference on*, pages 1344 –1349, aug. 2010.

[145] M. Steinbach, G. Karypis, and V. Kumar. A comparison of document clustering techniques. In *KDD Workshop on Text Mining*, 2000.

[146] Josef Steinberger, Mijail Kabadjov, Ralf Steinberger, Hristo Tanev, Marco Turchi, and Vanni Zavarella. Jrc's participation at tac 2011: Guided and multilingual summarization tasks. In *Proceedings of Text Analysis Conference (TAC 2011)*, 2011.

[147] N. Stolba and A. M. Tjoa. The relevance of data warehousing and data mining in the field of evidence-based medicine to support healthcare decision making. *International Journal of Computer Systems Science and Engineering*, 3(3):143–149, 2002.

[148] Chao-Ton Su, Pa-Chun Wang, Yan-Cheng Chen, and Li-Fei Chen. Data mining techniques for assisting the diagnosis of pressure ulcer development in surgical patients. *Journal of Medical Systems*, 36:2387–2399, 2012. 10.1007/s10916-011-9706-1.

[149] Laszlo Szathmary. *Symbolic Data Mining Methods with the Coron Platform*. PhD Thesis in Computer Science, University Henri PoincarÃ© – Nancy 1, France, Nov 2006.

[150] G. Taguchi, S. Chowdhury, and Y. Wu. *The Mahalanobis-Taguchi System*. McGraw-Hill, 2000.

[151] Hiroya Takamura and Manabu Okumura. Text summarization model based on the budgeted median problem. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 1589–1592, New York, NY, USA, 2009. ACM.

[152] Pang-Ning Tan, Vipin Kumar, and Jaideep Srivastava. Selecting the right interestingness measure for association patterns. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 32–41, New York, NY, USA, 2002. ACM.

[153] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining, (2nd Edition)*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2 edition, 2006.

[154] Murat Caner Testik, Banu Yuksel Ozkaya, Salih Aksu, and Osman Ilhami Ozcebe. Discovering blood donor arrival patterns using data mining: A method to investigate service quality at blood centers. *J. Med. Syst.*, 36(2):579–594, apr 2012.

[155] TexLexAn. Texlexan: An open-source text summarizer, 2011. Software available at *http://texlexan.sourceforge.net*.

[156] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe. Predicting elections with twitter: What 140 characters reveal about political sentiment. In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media*, pages 178–185, 2010.

[157] Stephanie M. van Rooden, Willem J. Heiser, Joost N. Kok, Dagmar Verbaan, Jacobus J. van Hilten, and Marinus Johan. The identification of parkinson's disease subtypes using cluster analysis: A systematic review. *Mov Disord*, 25(8):969–978, 2010.

[158] Vladimir N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[159] Jorge Vivaldi, Iria da Cunha, Juan Manuel Torres-Moreno, and Patricia VelÃ¡zquez-Morales. Automatic summarization using terminological and semantic resources. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may 2010. European Language Resources Association (ELRA).

[160] Thomas T. Wan. Healthcare informatics research: From data to evidence-based management. *J. Med. Syst.*, 30(1):3–7, February 2006.

[161] Dingding Wang and Tao Li. Document update summarization using incremental hierarchical clustering. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 279–288, New York, NY, USA, 2010. ACM.

[162] Dingding Wang, Shenghuo Zhu, Tao Li, Yun Chi, and Yihong Gong. Integrating document clustering and multidocument summarization. *ACM Trans. Knowl. Discov. Data*, 5(3):14:1–14:26, August 2011.

[163] Jianyong Wang and Jiawei Han. Bide: Efficient mining of frequent closed sequences. In *ICDE '04: Proceedings of the 20th International*

Conference on Data Engineering, pages 79–90, Washington, DC, USA, 2004. IEEE Computer Society.

[164] Xuewei Wang, Haibin Qu, Ping Liu, and Yiyu Cheng. A self-learning expert system for diagnosis in traditional chinese medicine. *Expert Systems with Applications*, 26(4):557 – 566, 2004.

[165] Anita Wasilewska. Cse634: Data mining concepts and techniques, 2006. CSE634 - Data Mining: Text Mining (Student's presentation). Accessed on 22 February 2013.
Course page: *http://www.cs.sunysb.edu/graduate/courses/cse634.html*
Slides: *http://www.cs.sunysb.edu/∼cse634/presentations/TextMining.pdf*.

[166] Johanna I. Westbrook, Enrico W. Coiera, A. Sophie Gosling, and Jeffrey Braithwaite. Critical incidents and journey mapping as techniques to evaluate the impact of online evidence retrieval systems on health care delivery and patient outcomes. *I. J. Medical Informatics*, pages 234–245, 2007.

[167] Sidney J. Winawer. Colorectal cancer screening. *Best Practice & Research Clinical Gastroenterology*, 21(6):1031 – 1048, 2007. The Multidisciplinary Management of Gastrointestinal Cancer.

[168] Pak Chung Wong, Paul Whitney, and Jim Thomas. Visualizing association rules for text mining. In *Proceedings of the 1999 IEEE Symposium on Information Visualization*, INFOVIS '99, pages 120–, Washington, DC, USA, 1999. IEEE Computer Society.

[169] WordNet. Wordnet: A lexical datatbase for english, 2010.
Online: *http://wordnet.princeton.edu*.

[170] Zi Yang, Keke Cai, Jie Tang, Li Zhang, Zhong Su, and Juanzi Li. Social context summarization. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR '11, pages 255–264, New York, NY, USA, 2011. ACM.

[171] Wen Yao and Akhil Kumar. Integrating clinical pathways into cdss using context and rules: a case study in heart disease. In *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*, IHI '12, pages 611–620, New York, NY, USA, 2012. ACM.

[172] Mohammed J. Zaki and Benjarath Phoophakdee. Mirage: A framework for mining, exploring and visualizing minimal association rules.

Technical report, Computer Science Dept., Rensselaer Polytechnic Inst, 2003.

[173] Shichao Zhang, Feng Chen, Xindong Wu, Chengqi Zhang, and Ruili Wang. Mining bridging rules between conceptual clusters. *Applied Intelligence*, 36:108–118, 2012. 10.1007/s10489-010-0247-y.

[174] Zhaoman Zhong and Zongtian Liu. Ranking events based on event relation graph for a single document. *Information Technology Journal*, 9:174–178, 2010.

[175] Junyan Zhu, Can Wang, Xiaofei He, Jiajun Bu, Chun Chen, Shujie Shang, Mingcheng Qu, and Gang Lu. Tag-oriented document summarization. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 1195–1196, New York, NY, USA, 2009. ACM.